James D. Veltrop
Jeremy C. Lowe
AXINN, VELTROP & HARKRIDER LLP
90 State House Square
Hartford, CT 06103
Tel: (860) 275-8100

James P. Doyle
AXINN, VELTROP & HARKRIDER LLP
114 West 47th Street
New York, NY 10036
Tel: (212) 728-2200

Attorneys for Plaintiff
PRODUCTION RESOURCE GROUP, L.L.C.

FILED
JUL 15 2008
USDC WP SDNY

08 CIV. 6333

JUDGE KARAS

UNITED STATES DISTRICT COURT
SOUTHERN DISTRICT OF NEW YORK

PRODUCTION RESOURCE GROUP, L.L.C., Plaintiff, v. MARTIN PROFESSIONAL, A/S and MARTIN PROFESSIONAL, INC., Defendants.	Civil Action No. _________ **COMPLAINT AND JURY DEMAND**

Plaintiff Production Resource Group, L.L.C. ("Plaintiff" or "PRG"), by its undersigned attorneys, for its Complaint herein against Defendants Martin Professional, A/S and Martin Professional, Inc. (collectively "Defendants" or "Martin"), alleges upon knowledge with respect to its own acts, and upon information and belief as to other matters, as follows:

THE PARTIES

1. Plaintiff PRG is a Delaware limited liability company having its principal place of business at 539 Temple Hill Road, New Windsor, New York 12553. PRG is in the business of, among other things, designing, manufacturing and selling sophisticated lighting fixtures, controllers and associated products for use in entertainment and display environments.

2. Defendant Martin Professional A/S is a Danish corporation having its principal place of business at Olof Palmes Allé 18, DK-8200 Åarhus N, Denmark.

3. Defendant Martin Professional, Inc. is a Florida corporation having its principal place of business at 700 Sawgrass Corporate Parkway, Sunrise, Florida 33325. Martin Professional, Inc. is a wholly-owned subsidiary of Martin Professional A/S and a United States distributor of products sold by Martin Professional A/S.

NATURE OF THE ACTION

4. This action arises under the patent laws of the United States, 35 U.S.C. § 1, *et seq*.

JURISDICTION AND VENUE

5. This Court has jurisdiction over the claims asserted herein pursuant to 28 U.S.C. §§ 1331 and 1338(a).

6. This Court has personal jurisdiction over Martin because Martin regularly transacts business within this judicial district and has committed acts of patent infringement within this judicial district.

7. Venue is proper in this Court pursuant to 28 U.S.C. §§ 1391(b) and (c) and 1400(b).

COUNT ONE
INFRINGEMENT OF UNITED STATES PATENT NO. 5,414,328

8. PRG incorporates each of the preceding paragraphs of this Complaint as if fully set forth herein.

9. PRG exclusively owns United States Patent No. 5,414,328 ("the '328 patent"), entitled "Stage Lighting Control Console Including Assignable Macro Functions," issued on May 9, 1995. See Exhibit A.

10. In violation of 35 U.S.C. § 271, Martin has made, used, offered for sale or sold within the United States or imported into the United States products that infringe one or more claims of the '328 patent, including the Maxxyz products.

11. Martin's infringement has been with knowledge of the '328 patent and has been willful.

12. PRG has suffered and will continue to suffer damages and irreparable injuries unless Martin's infringement of the '328 patent is enjoined.

COUNT TWO
INFRINGEMENT OF UNITED STATES PATENT NO. 5,502,627

13. PRG incorporates each of the preceding paragraphs of this Complaint as if fully set forth herein.

14. PRG exclusively owns United States Patent No. 5,502,627 ("the '627 patent"), entitled "Stage Lighting Lamp Unit and Stage Lighting System Including Such Unit," issued on March 26, 1996. See Exhibit B.

15. In violation of 35 U.S.C. § 271, Martin has made, used, offered for sale or sold within the United States or imported into the United States products that infringe one or more claims of the '627 patent, including the Mac 2000 products.

16. Martin's infringement has been with knowledge of the '627 patent and has been willful.

17. PRG has suffered and will continue to suffer damages and irreparable injuries unless Martin's infringement of the '627 patent is enjoined.

COUNT THREE
INFRINGEMENT OF UNITED STATES PATENT NO. 5,769,531

18. PRG incorporates each of the preceding paragraphs of this Complaint as if fully set forth herein.

19. PRG exclusively owns United States Patent No. 5,769,531 ("the '531 patent"), entitled "Stage Lighting Lamp Unit and Stage Lighting System Including Such Unit," issued on June 23, 1998. See Exhibit C.

20. In violation of 35 U.S.C. § 271, Martin has made, used, offered for sale or sold within the United States or imported into the United States products that infringe one or more claims of the '531 patent, including the Mac 2000 products.

21. Martin's infringement has been with knowledge of the '531 patent and has been willful.

22. PRG has suffered and will continue to suffer damages and irreparable injuries unless Martin's infringement of the '531 patent is enjoined.

COUNT FOUR
INFRINGEMENT OF UNITED STATES PATENT NO. 5,921,659

23. PRG incorporates each of the preceding paragraphs of this Complaint as if fully set forth herein.

24. PRG exclusively owns United States Patent No. 5,921,659 ("the '659 patent"), entitled "Stage Lighting Lamp Unit and Stage Lighting System Including Such Unit," issued on July 13, 1999. See Exhibit D.

25. In violation of 35 U.S.C. § 271, Martin has made, used, offered for sale or sold within the United States or imported into the United States products that infringe one or more claims of the '659 patent, including the Mac 2000 products.

26. Martin's infringement has been with knowledge of the '659 patent and has been willful.

27. PRG has suffered and will continue to suffer damages and irreparable injuries unless Martin's infringement of the '659 patent is enjoined.

COUNT FIVE
INFRINGEMENT OF UNITED STATES PATENT NO. 5,969,485

28. PRG incorporates each of the preceding paragraphs of this Complaint as if fully set forth herein.

29. PRG exclusively owns United States Patent No. 5,969,485 ("the '485 patent"), entitled "User Interface for a Lighting System that Allows Geometric and Color Sets to Be Simply Reconfigured," issued on October 19, 1999. See Exhibit E.

30. In violation of 35 U.S.C. § 271, Martin has made, used, offered for sale or sold within the United States or imported into the United States products that infringe one or more claims of the '485 patent, including the Maxxyz products.

31. Martin's infringement has been with knowledge of the '485 patent and has been willful.

32. PRG has suffered and will continue to suffer damages and irreparable injuries unless Martin's infringement of the '485 patent is enjoined.

COUNT SIX
INFRINGEMENT OF UNITED STATES PATENT NO. 6,011,662

33. PRG incorporates each of the preceding paragraphs of this Complaint as if fully set forth herein.

34. PRG exclusively owns United States Patent No. 6,011,662 ("the '662 patent"), entitled "Custom Color Wheel," issued on January 4, 2000. See Exhibit F.

35. In violation of 35 U.S.C. § 271, Martin has made, used, offered for sale or sold within the United States or imported into the United States products that infringe one or more claims of the '662 patent, including the Mac 700 and smartMac products.

36. Martin's infringement has been with knowledge of the '662 patent and has been willful.

37. PRG has suffered and will continue to suffer damages and irreparable injuries unless Martin's infringement of the '662 patent is enjoined.

COUNT SEVEN
INFRINGEMENT OF UNITED STATES PATENT NO. 6,029,122

38. PRG incorporates each of the preceding paragraphs of this Complaint as if fully set forth herein.

39. PRG exclusively owns United States Patent No. 6,029,122 ("the '122 patent"), entitled "Tempo Synchronization System for a Moving Light Assembly," issued on February 22, 2000. See Exhibit G.

40. In violation of 35 U.S.C. § 271, Martin has made, used, offered for sale or sold within the United States or imported into the United States products that infringe one or more claims of the '122 patent, including the Maxxyz products.

41. Martin's infringement has been with knowledge of the '122 patent and has been willful.

42. PRG has suffered and will continue to suffer damages and irreparable injuries unless Martin's infringement of the '122 patent is enjoined.

COUNT EIGHT
INFRINGEMENT OF UNITED STATES PATENT NO. 6,062,706

43. PRG incorporates each of the preceding paragraphs of this Complaint as if fully set forth herein.

44. PRG exclusively owns United States Patent No. 6,062,706 ("the '706 patent"), entitled "Variable Color Fluorescent Lighting," issued on May 16, 2000. See Exhibit H.

45. In violation of 35 U.S.C. § 271, Martin has made, used, offered for sale or sold within the United States or imported into the United States products that infringe one or more claims of the '706 patent, including the Cyclo Directional products.

46. Martin's infringement has been with knowledge of the '706 patent and has been willful.

47. PRG has suffered and will continue to suffer damages and irreparable injuries unless Martin's infringement of the '706 patent is enjoined.

COUNT NINE
INFRINGEMENT OF UNITED STATES PATENT NO. 6,256,136

48. PRG incorporates each of the preceding paragraphs of this Complaint as if fully set forth herein.

49. PRG exclusively owns United States Patent No. 6,256,136 ("the '136 patent"), entitled "Pixel Based Gobo Record Control Format," issued on July 3, 2001. See Exhibit I.

50. In violation of 35 U.S.C. § 271, Martin has made, used, offered for sale or sold within the United States or imported into the United States products that infringe one or more claims of the '136 patent, including the Maxedia products.

51. Martin's infringement has been with knowledge of the '136 patent and has been willful.

52. PRG has suffered and will continue to suffer damages and irreparable injuries unless Martin's infringement of the '136 patent is enjoined.

COUNT TEN
INFRINGEMENT OF UNITED STATES PATENT NO. 6,288,828

53. PRG incorporates each of the preceding paragraphs of this Complaint as if fully set forth herein.

54. PRG exclusively owns United States Patent No. 6,288,828 ("the '828 patent"), entitled "Programmable Light Beam Shape Altering Device Using Programmable Micromirrors," issued on September 11, 2001. See Exhibit J.

55. In violation of 35 U.S.C. § 271, Martin has made, used, offered for sale or sold within the United States or imported into the United States products that infringe one or more claims of the '828 patent, including the Maxedia products.

56. Martin's infringement has been with knowledge of the '828 patent and has been willful.

57. PRG has suffered and will continue to suffer damages and irreparable injuries unless Martin's infringement of the '828 patent is enjoined.

COUNT ELEVEN
INFRINGEMENT OF UNITED STATES PATENT NO. 6,326,741

58. PRG incorporates each of the preceding paragraphs of this Complaint as if fully set forth herein.

59. PRG exclusively owns United States Patent No. 6,326,741 ("the '741 patent"), entitled "Stage Lighting Lamp Unit and Stage Lighting System Including Such Unit," issued on December 4, 2001. See Exhibit K.

60. In violation of 35 U.S.C. § 271, Martin has made, used, offered for sale or sold within the United States or imported into the United States products that infringe one or more claims of the '741 patent, including the Mac 2000 products.

61. Martin's infringement has been with knowledge of the '741 patent and has been willful.

62. PRG has suffered and will continue to suffer damages and irreparable injuries unless Martin's infringement of the '741 patent is enjoined.

COUNT TWELVE
INFRINGEMENT OF UNITED STATES PATENT NO. 6,466,357

63. PRG incorporates each of the preceding paragraphs of this Complaint as if fully set forth herein.

64. PRG exclusively owns United States Patent No. 6,466,357 ("the '357 patent"), entitled "Pixel Based Gobo Record Control Format," issued on October 15, 2002. See Exhibit L.

65. In violation of 35 U.S.C. § 271, Martin has made, used, offered for sale or sold within the United States or imported into the United States products that infringe one or more claims of the '357 patent, including the Maxedia products.

66. Martin's infringement has been with knowledge of the '357 patent and has been willful.

67. PRG has suffered and will continue to suffer damages and irreparable injuries unless Martin's infringement of the '357 patent is enjoined.

COUNT THIRTEEN
INFRINGEMENT OF UNITED STATES PATENT NO. 6,515,435

68. PRG incorporates each of the preceding paragraphs of this Complaint as if fully set forth herein.

69. PRG exclusively owns United States Patent No. 6,515,435 ("the '435 patent"), entitled "Pixel Mirror Based Stage Lighting System," issued on February 4, 2003. See Exhibit M.

70. In violation of 35 U.S.C. § 271, Martin has made, used, offered for sale or sold within the United States or imported into the United States products that infringe one or more claims of the '435 patent, including the Mac 2000 products.

71. Martin's infringement has been with knowledge of the '435 patent and has been willful.

72. PRG has suffered and will continue to suffer damages and irreparable injuries unless Martin's infringement of the '435 patent is enjoined.

COUNT FOURTEEN
INFRINGEMENT OF UNITED STATES PATENT NO. 6,549,326

73. PRG incorporates each of the preceding paragraphs of this Complaint as if fully set forth herein.

74. PRG exclusively owns United States Patent No. 6,549,326 ("the '326 patent"), entitled "Pixel Based Gobo Record Control Format," issued on April 15, 2003. See Exhibit N.

75. In violation of 35 U.S.C. § 271, Martin has made, used, offered for sale or sold within the United States or imported into the United States products that infringe one or more claims of the '326 patent, including the Maxedia products.

76. Martin's infringement has been with knowledge of the '326 patent and has been willful.

77. PRG has suffered and will continue to suffer damages and irreparable injuries unless Martin's infringement of the '326 patent is enjoined.

COUNT FIFTEEN
INFRINGEMENT OF UNITED STATES PATENT NO. 6,597,132

78. PRG incorporates each of the preceding paragraphs of this Complaint as if fully set forth herein.

79. PRG exclusively owns United States Patent No. 6,597,132 ("the '132 patent"), entitled "Stage Lighting Lamp Unit and Stage Lighting System Including Such Unit," issued on July 22, 2003. See Exhibit O.

80. In violation of 35 U.S.C. § 271, Martin has made, used, offered for sale or sold within the United States or imported into the United States products that infringe one or more claims of the '132 patent, including the Mac 2000 products.

81. Martin's infringement has been with knowledge of the '132 patent and has been willful.

82. PRG has suffered and will continue to suffer damages and irreparable injuries unless Martin's infringement of the '132 patent is enjoined.

COUNT SIXTEEN
INFRINGEMENT OF UNITED STATES PATENT NO. 6,891,656

83. PRG incorporates each of the preceding paragraphs of this Complaint as if fully set forth herein.

84. PRG exclusively owns United States Patent No. 6,891,656 ("the '656 patent"), entitled "Pixel Based Gobo Record Control Format," issued on May 10, 2005. See Exhibit P.

85. In violation of 35 U.S.C. § 271, Martin has made, used, offered for sale or sold within the United States or imported into the United States products that infringe one or more claims of the '656 patent, including the Maxedia products.

86. Martin's infringement has been with knowledge of the '656 patent and has been willful.

87. PRG has suffered and will continue to suffer damages and irreparable injuries unless Martin's infringement of the '656 patent is enjoined.

COUNT SEVENTEEN
INFRINGEMENT OF UNITED STATES PATENT NO. 6,894,443

88. PRG incorporates each of the preceding paragraphs of this Complaint as if fully set forth herein.

89. PRG exclusively owns United States Patent No. 6,894,443 ("the '443 patent"), entitled "Stage Lighting Lamp Unit and Stage Lighting System Including Such Unit," issued on May 17, 2005. See Exhibit Q.

90. In violation of 35 U.S.C. § 271, Martin has made, used, offered for sale or sold within the United States or imported into the United States products that infringe one or more claims of the '443 patent, including the Mac 2000 products.

91. Martin's infringement has been with knowledge of the '443 patent and has been willful.

92. PRG has suffered and will continue to suffer damages and irreparable injuries unless Martin's infringement of the '443 patent is enjoined.

COUNT EIGHTEEN
INFRINGEMENT OF UNITED STATES PATENT NO. 6,934,071

93. PRG incorporates each of the preceding paragraphs of this Complaint as if fully set forth herein.

94. PRG exclusively owns United States Patent No. 6,934,071 ("the '071 patent"), entitled "Pixel Based Gobo Record Control Format," issued on August 23, 2005. See Exhibit R.

95. In violation of 35 U.S.C. § 271, Martin has made, used, offered for sale or sold within the United States or imported into the United States products that infringe one or more claims of the '071 patent, including the Maxxyz products.

96. Martin's infringement has been with knowledge of the '071 patent and has been willful.

97. PRG has suffered and will continue to suffer damages and irreparable injuries unless Martin's infringement of the '071 patent is enjoined.

COUNT NINETEEN
INFRINGEMENT OF UNITED STATES PATENT NO. 7,020,370

98. PRG incorporates each of the preceding paragraphs of this Complaint as if fully set forth herein.

99. PRG exclusively owns United States Patent No. 7,020,370 ("the '370 patent"), entitled "Three Color Digital Gobo System," issued on March 28, 2006. See Exhibit S.

100. In violation of 35 U.S.C. § 271, Martin has made, used, offered for sale or sold within the United States or imported into the United States products that infringe one or more claims of the '370 patent, including the Mac 2000 products.

101. Martin's infringement has been with knowledge of the '370 patent and has been willful.

102. PRG has suffered and will continue to suffer damages and irreparable injuries unless Martin's infringement of the '370 patent is enjoined.

COUNT TWENTY
INFRINGEMENT OF UNITED STATES PATENT NO. 7,181,112

103. PRG incorporates each of the preceding paragraphs of this Complaint as if fully set forth herein.

104. PRG exclusively owns United States Patent No. 7,181,112 ("the '112 patent"), entitled "Three Color Digital Gobo System," issued on February 20, 2007. See Exhibit T.

105. In violation of 35 U.S.C. § 271, Martin has made, used, offered for sale or sold within the United States or imported into the United States products that infringe one or more claims of the '112 patent, including the Mac 2000 products.

106. Martin's infringement has been with knowledge of the '112 patent and has been willful.

107. PRG has suffered and will continue to suffer damages and irreparable injuries unless Martin's infringement of the '112 patent is enjoined.

PRAYER FOR RELIEF

WHEREFORE, PRG respectfully requests that this Court enter judgment in its favor and against Martin and grant the following relief:

A. A judgment that Martin has infringed the '328, '627, '531, '659, '485, '662, '122, '706, '136, '828, '741, '357, '435, '326, '132, '656, '443, '071, '370 and '112 patents directly, or by inducement or contribution, in violation of 35 U.S.C. § 271;

B. A judgment that Martin's infringement of the '328, '627, '531, '659, '485, '662, '122, '706, '136, '828, '741, '357, '435, '326, '132, '656, '443, '071, '370 and '112 patents has been willful;

C. An order, pursuant to 35 U.S.C. § 283, enjoining Martin and all persons in active concert or participation with Martin from any further infringement of the '328, '627, '531, '659, '485, '662, '122, '706, '136, '828, '741, '357, '435, '326, '132, '656, '443, '071, '370 and '112 patents;

D. An order, pursuant to 35 U.S.C. § 284, awarding PRG damages adequate to compensate for Martin's infringement of the '328, '627, '531, '659, '485, '662, '122, '706, '136, '828, '741, '357, '435, '326, '132, '656, '443, '071, '370 and '112 patents, in amounts to be determined at trial, but in no event less than reasonable royalties;

E. An order, pursuant to 35 U.S.C. § 284, trebling all damages awarded to PRG in view of Martin's willful infringement of the '328, '627, '531, '659, '485, '662, '122, '706, '136, '828, '741, '357, '435, '326, '132, '656, '443, '071, '370 and '112 patents;

F. An order, pursuant to 28 U.S.C. § 1961 and 35 U.S.C. § 284, awarding to PRG interest on the damages and its costs incurred from this action;

G. An order, pursuant to 35 U.S.C. § 285, awarding to PRG its attorneys' fees;

H. An order directing Martin to recall from distribution and destroy its entire stock of infringing products within the United States; and

I. Such other and further relief as the Court may deem just and proper.

JURY DEMAND

In accordance with Fed. R. Civ. P. 38 and 39, Plaintiff PRG asserts its right under the Seventh Amendment to the United States Constitution and demands a trial by jury on all issues that may be so tried.

Dated: July 15, 2008

Respectfully submitted,

James D. Veltrop
Jeremy C. Lowe
AXINN, VELTROP & HARKRIDER LLP
90 State House Square
Hartford, CT 06103
Tel: (860) 275-8100
Fax: (860) 275-8101

James P. Doyle
AXINN, VELTROP & HARKRIDER LLP
114 West 47th Street
New York, NY 10036
Tel: (212) 728-2200
Fax: (212) 728-2201

Attorneys for Plaintiff
PRODUCTION RESOURCE GROUP, L.L.C.

EXHIBIT A

US005414328A

United States Patent [19]

Hunt et al.

[11] Patent Number: 5,414,328

[45] Date of Patent: May 9, 1995

[54] STAGE LIGHTING CONTROL CONSOLE INCLUDING ASSIGNABLE MACRO FUNCTIONS

[75] Inventors: Mark A. Hunt, Derby; Keith J. Owen, Moseley, United Kingdom

[73] Assignee: Light & Sound Design, Ltd., Edinburgh, United Kingdom

[21] Appl. No.: 77,862

[22] Filed: Jun. 18, 1993

[30] Foreign Application Priority Data

Nov. 19, 1992 [GB] United Kingdom 9224287
Apr. 20, 1993 [GB] United Kingdom 9308070

[51] Int. Cl.[6] H05B 37/00
[52] U.S. Cl. 315/316; 315/312; 315/314; 315/292
[58] Field of Search .... 315/312, 314, 316, 292, 315/294, 324; 362/85, 233; 340/324 A

[56] References Cited

U.S. PATENT DOCUMENTS

3,706,914	12/1972	Van Buren	315/316
3,898,643	8/1975	Ettlinger	315/316 X
4,716,344	12/1987	Newell et al.	315/312
4,772,825	9/1988	Tabor et al.	315/312
4,792,731	12/1988	Pearlman et al.	315/316
5,010,459	4/1991	Taylor et al.	315/316 X
5,059,871	10/1991	Pearlman et al.	315/312 X

FOREIGN PATENT DOCUMENTS

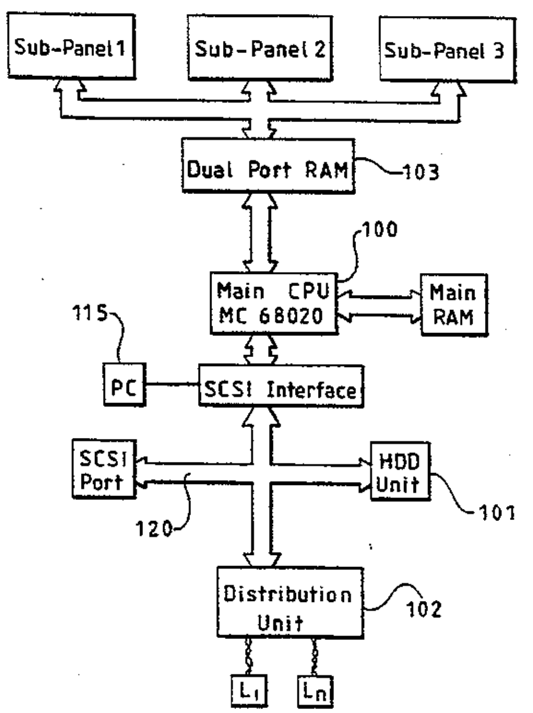
0495305	7/1992	European Pat. Off. .
2231138	11/1990	United Kingdom .
2235310	2/1991	United Kingdom .
WO89/05086	6/1989	WIPO .
WO89/05421	6/1989	WIPO .

Primary Examiner—Robert J. Pascal
Assistant Examiner—Haissa Philogene
Attorney, Agent, or Firm—Fish & Richardson

[57] ABSTRACT

A control console for controlling stage lighting includes a panel on which there are mounted a multiplicity of control elements in the form of switches, slider potentiometers and others which can be used by an operator to input control data to the console and directly control remote lamp units connected to the console. The console includes a main cpu which forms part of an electronic control system which includes function allocation means enabling the operator to determine which one of the control elements exercises control over a particular independently controllable function of one or more of the lamps. The console also includes a data distribution unit which includes a plurality of separate serial communication controllers for communicating with respective ones of the lamps. An SCSI bus is used for transferring data between the main cpu and the distribution unit, which includes its own processor unit for distributing the data to the serial communication controllers.

17 Claims, 4 Drawing Sheets



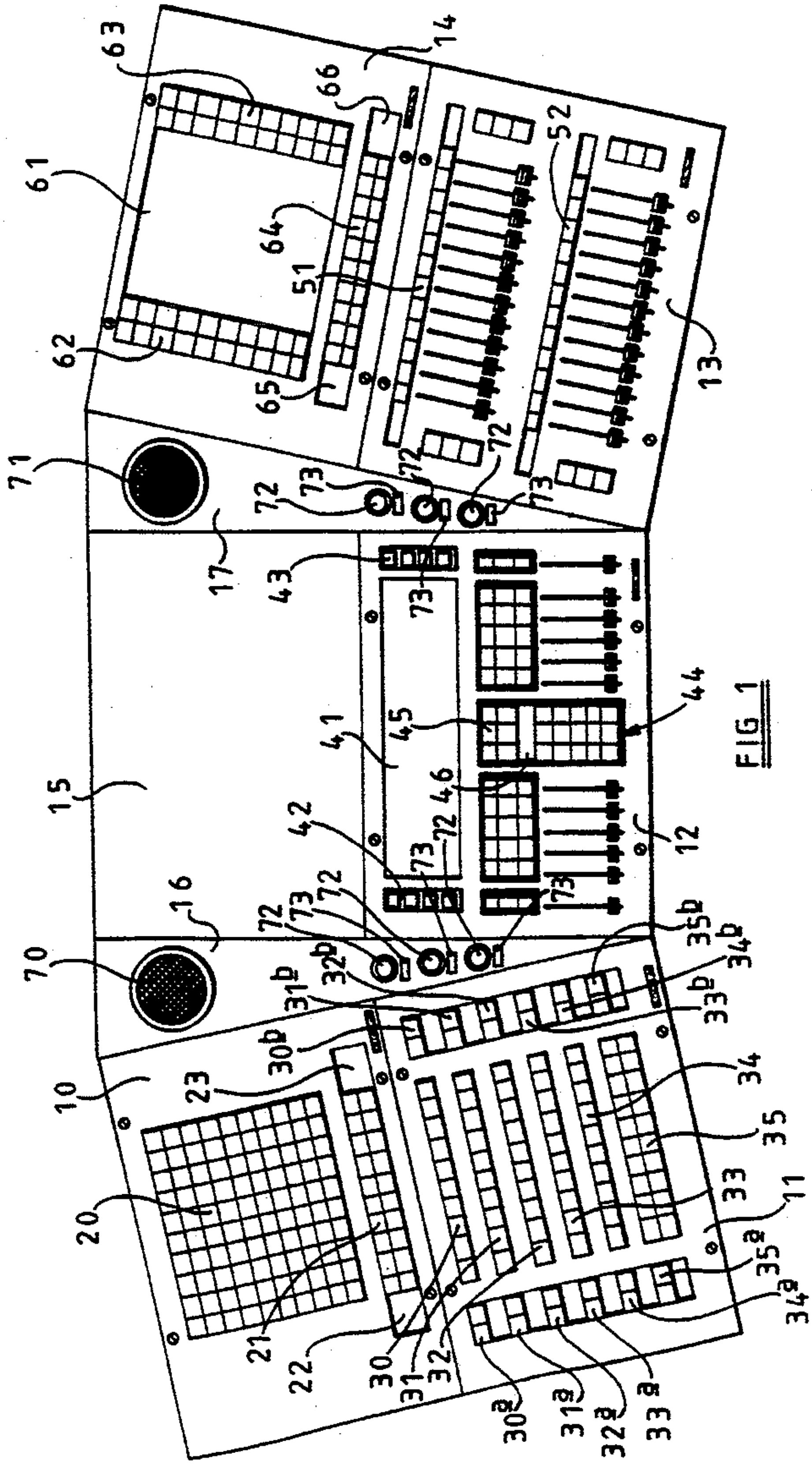


FIG 1

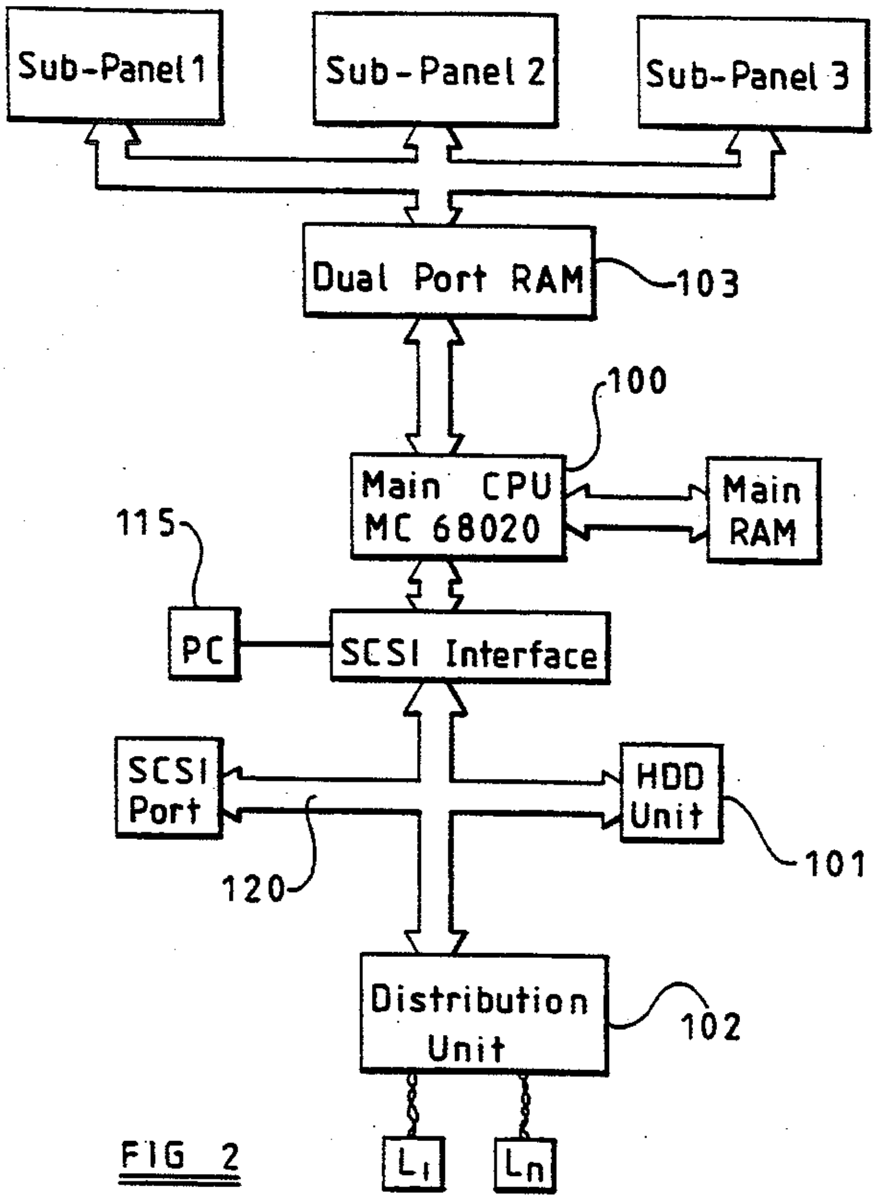
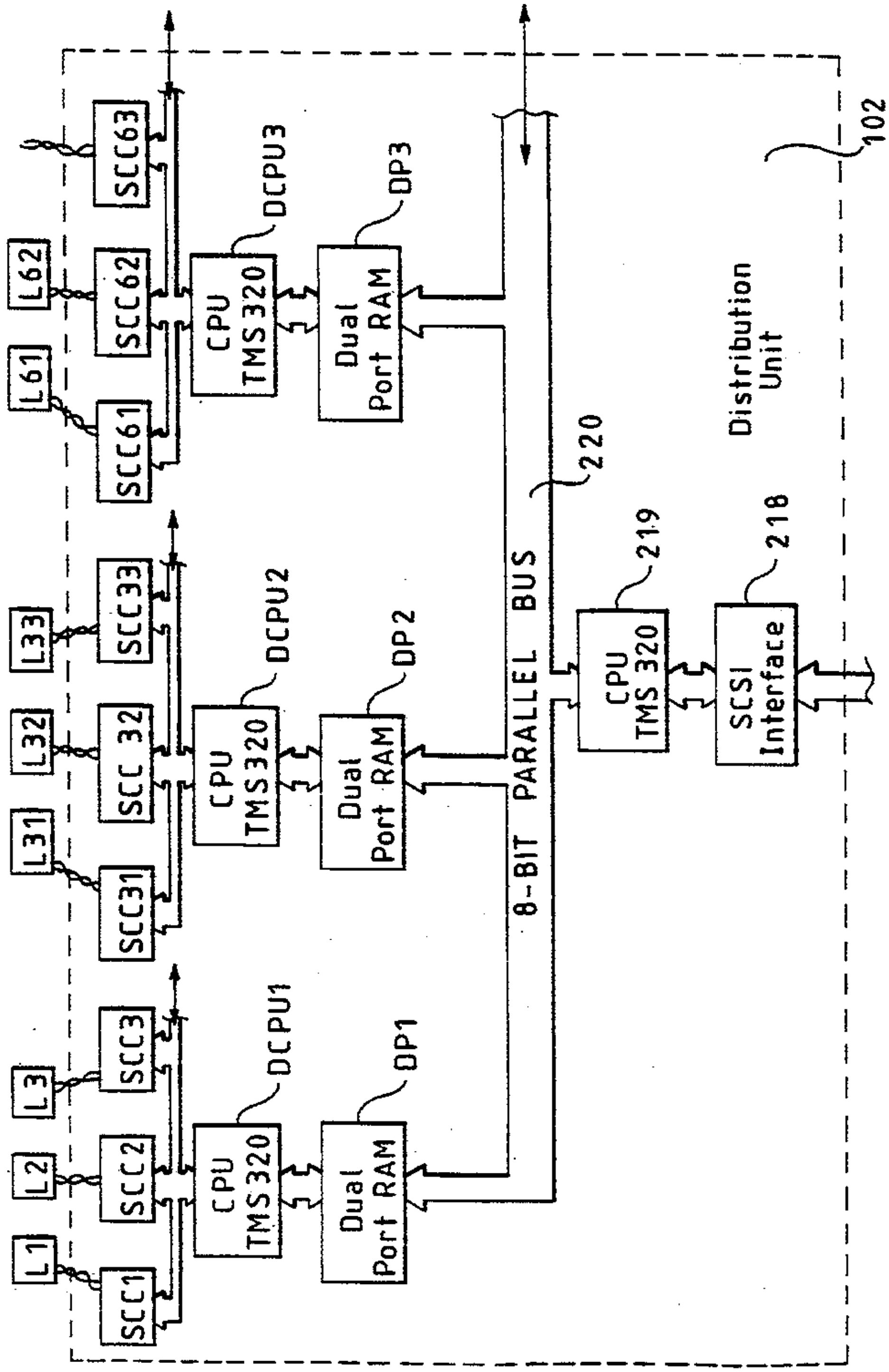
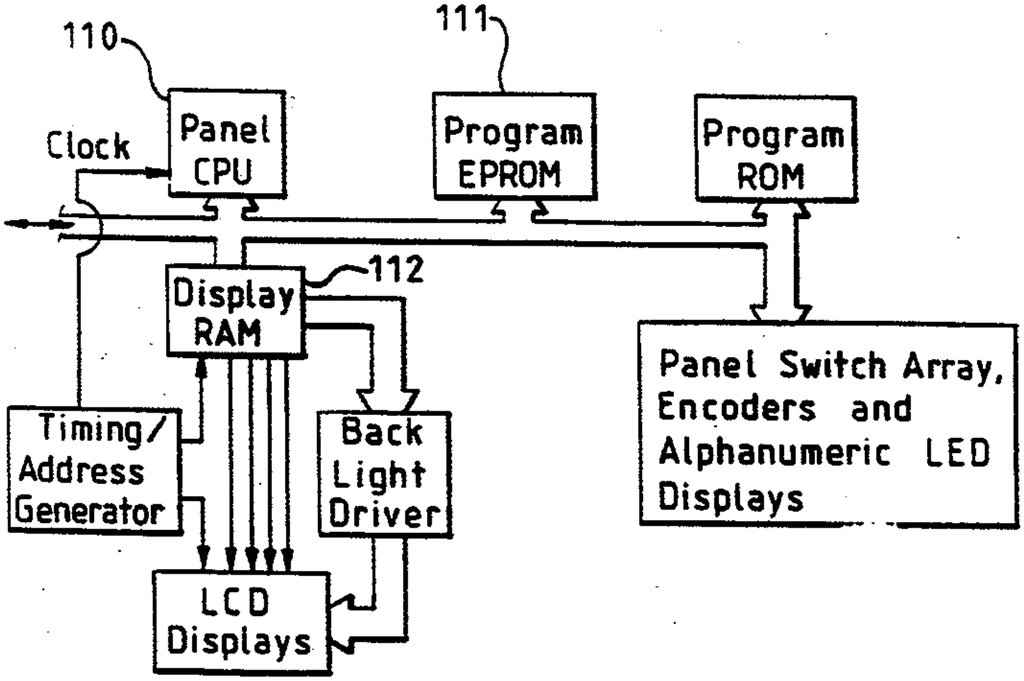

Sub-Panel 1
Sub-Panel 2
Sub-Panel 3
Dual Port RAM
103
100
Main CPU
MC 68020
Main
RAM
115
PC
SCSI Interface
SCSI
Port
HDD
Unit
101
120
Distribution
Unit
102
L1
Ln

FIG 2



L1
L2
L3
SCC1
SCC2
SCC3
CPU TMS320
DCPU1
Dual Port RAM
DP1
L31
L32
L33
SCC31
SCC 32
SCC33
CPU TMS320
DCPU2
Dual Port RAM
DP2
L61
L62
SCC61
SCC62
SCC63
CPU TMS320
DCPU3
Dual Port RAM
DP3
8-BIT PARALLEL BUS
220
CPU TMS 320
219
SCSI Interface
218
Distribution Unit
102

FIG 3



110
111
Panel CPU
Program EPROM
Program ROM
Clock
112
Display RAM
Panel Switch Array, Encoders and Alphanumeric LED Displays
Timing/ Address Generator
Back Light Driver
LCD Displays

FIG 4

STAGE LIGHTING CONTROL CONSOLE INCLUDING ASSIGNABLE MACRO FUNCTIONS

FIELD OF THE INVENTION

This invention relates to a console for controlling a stage lighting system.

BACKGROUND AND SUMMARY OF THE INVENTION

Conventional stage lighting control consoles make use of a plurality of control elements arranged on a panel and each controlling a specific function of the console. The control elements may include push-button or multi-position switches, and rotary and linear motion potentiometers. The designer of the console decides during the design operation exactly what lighting function each switch or potentiometer will control and the operator of the console has very little choice about how he uses the console.

Modern stage lighting has become more and more complex in recent years and consoles are required to control a large number of different functions. A lighting unit may, for example, have built in systems for remote control of intensity, beam direction, beam spread, colour variation, gobo positioning and rotation and other functions. Lighting designers require to make use of all these functions in different combinations and it has therefore become very difficult to provide an ergonomic console layout which will satisfy all the lighting designers varying needs.

It is therefore an object of the present invention to provide a control console which is capable of being readily reconfigured to allow it to be used in a variety of different ways.

In accordance with the invention there is provided a stage lighting control console including a panel, a plurality of control elements mounted on said panel and including manually actuable switch elements and manually adjustable control elements and an electronic control system controlled by said switch element and manually adjustable control elements for producing output signals for controlling a multiplicity of independently controllable functions of a plurality of remote lamp units, said electronic control system including function allocation means for determining which of the switch elements and manually adjustable control elements exercise control over each one of the functions of the lamp units, said function allocation means being under the control of a user of the console utilizing selected ones of the switch elements.

The individual switch elements may incorporate in built display devices which are controlled by the electronic control system to provide a display appropriate to the current function of the switch element.

The panel may also include a plurality of further display areas adjacent individual switch elements or control elements or adjacent groups of such element to display alphanumeric information indicating the current function of the associated switch or control element or the associated group of such elements.

Preferably, the panel comprises a plurality of sub-panels each with a different arrangement of switch elements, control elements and display areas, each sub-panel having its own associated processor unit for controlling the display areas of that sub-panel and the console including a main processor unit which communicates with the sub-panel processor units.

With this arrangement, the user is able to determine for himself the relative locations of the sub-panels.

Preferably, there is provided a dual-port memory unit which is connected to the main processor unit and to all the sub-panel processor units, such memory unit being used to store switch and control element operation data and display data, the main processor unit reading from the memory unit the switch and control element operation data and writing to the memory unit display data, and the sub-panel processor units writing switch and control element operation data to the memory unit and reading therefrom the display data.

The invention also resides in a stage lighting console comprising the combination of a panel, a plurality of user input control elements on said panel, a main processor unit for processing data input by a user to generate a multiplicity of data message blocks to be sent respectively to a multiplicity of individually controllable lamp units, and a distribution unit including a multiplicity of individual serial communication controllers for connection to respective ones of the lamp units, first data bus means connecting said main processor unit to said distributor unit for transferring said message data blocks to said distribution unit and second processor means for distributing the data contained in said message data blocks to said individual serial communication controllers.

Preferably, the main processor unit has main RAM in a portion of which it stores said message data blocks, and said distribution unit has a plurality of blocks of dual port RAM into which the contents of said portion of the main RAM can be copied over said first data bus under the control of said second processor means.

For controlling transfer of data from the blocks of dual port RAM to the associated serial communication controllers, there are preferably a plurality of third processors units each associated with a different one of the blocks of dual port RAM and with a different group of the serial communication controllers.

Each of said third processor units is programmed to transfer data from the associated block of dual port RAM to the serial communication controllers of the associated group in an interleaved byte-by-byte fashion, each serial communication controller transmitting each byte on receipt thereof.

BRIEF DESCRIPTION OF THE DRAWINGS

An example of the invention is shown in the accompanying drawings, in which:

FIG. **1** is a diagrammatic view of the panel of the console;

FIG. **2** is a schematic diagram of the control system;

FIG. **3** is a schematic diagram of a distribution unit which is shown in FIG. **2**; and

FIG. **4** is a schematic diagram of the control system for a single subpanel of the panel shown in FIG. **1**.

DESCRIPTION OF THE PREFERRED EMBODIMENT

The panel shown in FIG. **1** includes five separate sub-panels **10**, **11**, **12**, **13** and **14**, together with a blank sub-panel **15** and two wedge shaped panel pieces **16** and **17**.

Mounted on the various sub-panels are a variety of switch elements, manually operable control elements,

(such as slider-type potentiometers and rotary encoders) and a plurality of display areas.

Each of the switch elements used is of the same type incorporating a push-button with an inset dot matrix LCD display capable of displaying numbers, alphabetic characters and graphic symbols and back lighting LED's. Red and Green LED's are incorporated and these can be illuminated at different levels so as to enable the switches to be illuminated at different levels in different colours in the red-yellow-green colour range as required. The display areas make use of multi-segment LED alphanumeric display elements.

Sub-panel **10**, which is used for selecting the lamp channel or channels to be controlled by subsequent actions, has a 10×10 bank **20** of the switch elements and two additional rows of ten switch elements. The switches in the 10×10 bank are arranged to display symbols, such as the numbers between 1 and 100. The sub-panel **10** also includes two of the display areas **22, 23** at opposite ends of double row of switch elements. Each of these display areas can display two rows of eight characters.

The sub-panel **11** has five single rows **30** to **34** of ten of the switch elements and a double row **35** of switch elements. Aligned with each of the five rows **30** to **34** are pairs of the switch elements **30**_^{a}, **30**_^{b} to **34**_^{a}, **34**_^{b} at opposite sides of the sub-panel and two more pairs of switch elements **35**_^{a} and **35**_^{b} are similarly associated with the double row **35**. **34**^{b} at opposite sides of the sub-panel and two more pairs of switch Thus there are twelve of these pairs of switches altogether and there is a separate eight-character display area adjacent each switch pair. FIG. **1** shows these display areas arranged beneath respective switch pairs, but each display area is preferably above the associated switch pair so that the display area is non-concealed by the hand of the user operating the switch elements.

Sub-panel **12** includes a display area **41** fifty six characters wide by four characters high. Vertical rows **42, 43** of four switch elements are arranged at opposite sides of this area **41**. In the centre of sub-panel **12** there is a numeric key pad **44** made up of a five by three array of the switch elements. Between this and the area **41** is a double row **45** of three switch elements and a twelve character wide single row display area **46**. On each side of the numeric key-pad is an array of six slider potentiometers and a six by three array of the switch elements.

Sub-panel **13** has two rows **51, 52** of twelve of switch elements, with two rows of twelve slider potentiometers below the respective rows of switch elements. There are four vertical rows of three switch elements. At the ends of each of the rows of twelve switch elements there are display areas each providing an eight character display.

Finally, sub-panel **14**, includes a large display area **61** which can display ten rows of 28 characters. A single or (as shown) double vertical row **62, 63** of ten switch elements is arranged at each side of area **61** and there is a double horizontal row **64** of switch elements below the area **61**, with a two row by eight character display area **65, 66** at each end.

In the two wedge panel portions **16** and **17** there are provided a pair of loudspeakers **70, 71**, a number of rotary digital encoders **72** and a like number of alphanumeric LED display areas **73** adjacent respective encoders.

Referring now to FIG. **2**, the electronic control system shown includes a main CPU **100** which accepts input from the switch elements, potentiometers and encoders, as well as from a hard disk drive **101** on an SCSI bus which is also connected to an SCSI port, to which, if required, a personal computer **115** having an SCSI port can be connected for programming the system, reading from or writing to the hard disk unit, or taking over from the main cpu **100** in cases of failure of the console. The SCSI bus is also connected to a distribution unit **102** shown in more detail in FIG. **3**.

The main console cpu **100** creates messages to be sent to the individual lamps, each message comprising a fixed number of bytes for each lamp. The messages contain data relating to the parameters of the lamp, which parameters can include required lamp orientation, beam coloration, iris diaphragm diameter, gobo selection and rotation, zoom projection lens control and opening or closing of a shutter included in the lamp. A block of the RAM of the main cpu is set aside for the storage of these messages, the block being large enough to contain messages for **240** lamps, being the largest number which can be controlled via the distribution unit. Where it is required to control more than **240** lamps additional distribution units can be connected to the SCSI bus and extra main cpu RAM reserved for message storage. When any message data is changed the main cpu **100** sets a flag in the RAM block which is detected at a given point in the main cpu program loop and interpreted as a signal that the changed message data is to be transferred to the distribution unit **102**.

The distribution unit **102** has a main cpu **219** which controls reception of data from the SCSI bus interface and distribution of such data to up to eight blocks of dual,port memory **DP1**, **DP2**, **DP3** . . . via an eight bit data bus **220**. The cpu **219** is alerted to the waiting message data when cpu **101** selects the distribution unit. The cpu **219** then supervises byte by byte transfer of the message data which it routes to the various blocks of dual port memory.

For actually sending out the message data to the lamps, there are a plurality of serial communication controllers **SCC1** to **SCC30**, **SCC31** to **SCC60** etc, there being thirty serial communication controllers associated with each block of dual port memory. A further cpu **DCPU1**, **DCPU2**, etc is associated with each block of dual port memory and distributes message data transferred to the dual port memory to the individual serial communication controllers and the messages are transferred to the lamps. Each serial communication controller in the distribution unit includes a line driver which is disabled except when data is to be transmitted. Enabling of the driver can cause a spurious signal to be transmitted over the data link. To allow such spurious signals to be identified and ignored, a two-byte gap is left between enabling the line driver and commencing transmission of the message data for the channel in question. This is described in more detail in the co-pending application of even date entitled "Stage Lighting Lamp Unit and Stage Lighting System Including Such Unit" of Hunt, Owen and Hughes, Ser. No. 08/077,877.

Each of the cpus eg **DCPU1**, transfers data from the associated dual port RAM **DP1** to the serial communication controller **SCC1** to **SCC30** with which it is associated one byte at a time, ie the first byte for **SCC1** is transferred followed by the first byte for **SCC2** and so on, each serial communication controller commencing transmission as soon as it has received its byte of data. The serial communication controllers operate to transmit data at 230.4 Kbps so that it takes about 35 μs to

transmit each byte. Transfer of data from the dual port RAM DP1 to the serial communication controllers is, however, at a rate of several Mbps, so that the transmissions from all the serial communication controllers are almost simultaneous. The cpu DCPU1 is not required to monitor the transmission of data by the serial communication controller, but utilizes a software timer to commence transfer of the second byte to the serial communication controllers. This timer is started when transfer of the byte of data to the last serial communication controller SCC30 has been completed and its time-out duration is slightly longer than the byte transmission time, say 40 μs. Transmission of all the messages takes about 1.5 ms out of a distribution unit main program loop duration of 4 ms.

The main CPU also controls the LCD matrix displays in the various switch elements and the alphanumeric LED display areas of the panels.

As shown in FIG. **4**, each sub-panel has its own local CPU **110** which is connected by a data bus to a block of dual port RAM **103** (FIG. **2**) which is also connected to the main CPU. The dual port RAM **103**, has a range of addresses for each of the panels and, within the range reserved for each panel, there is stored both switch element and control element data to be conveyed from the panel to the main CPU and display data to be conveyed to the panel CPU. Each panel CPU has its own operating program stored in EPROM **111**, such program being matched to the configuration switch elements, control elements and display areas on the panel. The LCD matrix displays and their backlight LEDs are controlled by display RAM **112** which is refreshed periodically by the CPU **110**. A multiplexing arrangement is used to drive the LCD matrix elements and the backlight LCDs. The CPU **110** polls the various switch elements and control elements on its panel and writes data relating to the current status of these into the dual port RAM area reserved for this data. The CPU **110** also controls the LED alphanumeric display areas on its panel.

Those of ordinary skill in the stage lighting art understand that a cue represents a prestored set of parameters for a particular lamp. For example, cue number **10** for lamp number **12** might indicate that lamp number **12** should go to specified pan and tilt positions, and should be controlled to have specified values of iris, intensity, etc., whenever cue number **10** is executed. Cues must be recorded in advance: and sometimes this is done by setting the lamp manually to the desired parameters and executing a cue record command.

The main CPU **100** polls the dual port RAM periodically for key-press events and changes in the outputs of the control elements. The data received in this way used either to set up dialogs with the user concerning the control actions he wishes to take, or, if no dialog is necessary or the dialog is complete, to write data to the hard disk drive or send it out over the network.

The main CPU software includes a panel manager routine which is responsible for collecting all keypress events, wherever they occur on the sub-panels. Each area has its own rules/methods regarding keypress events.

The "select" area on sub-panel **10** is used to select which of the lamps will be influenced by the cue record being assembled and a selection manager module is passed all data related to keypress events in this area. The selection manager maintains its own map of which lamp channels are currently selected. Whenever one of the select keys is pressed the corresponding channel status in this map is changed. If the channel was previously selected it is deselected and vice versa. If one key in the 10×10 array is pressed and held and another key is then pressed, all the channels between those represented by the two keys are selected. If a particular key is "double-clicked" (i.e. pressed twice within a given time) the associated channel is selected and all others are deselected. The selection manager outputs channel status data to the dual port memory so that only the keys associated with selected channels are highlighted (i.e. illuminated bright yellow instead of low level red).

The numeric key pad area **44** is used to control a cue manager software module. Entry of a valid (non-zero) number on the key pad followed by a press of an "ENTER" key will make a cue record with that number the current cue, creating a new cue if one does not already exist. There are also "NEXT" and "LAST" keys which call up the next and previous cues respectively. None of these key actions executes the chosen cue, but merely makes it current. Pressing of a "GO" key causes the cue to be executed.

The sub-panel **11** contains the groups of switch elements which are the most likely to be reassigned to different tasks by different users. There are effectively six groups of keys in this area, each group containing two key pairs and ten (or twenty) keys in between the two pairs. The pairs and the row may be used individually or grouped together.

In the individual mode each of the two pairs of keys may control a different lamp function. Pressing the left hand key decreases the value assigned to that function, while pressing the right hand one increases the value. Holding down either key causes the function value to accelerate in the appropriate direction. Releasing the key stops the value changing. Double-clicking the key sets the value to its maximum or minimum. In the same mode, a central group of ten (or twenty) keys acts as a group of eight (or eighteen) preset value keys, with the two left hand keys acting as scroll keys, so that more than eight (or eighteen) preset values can be accessed.

In the combined mode, the row(s) are all preset select keys, while the right hand key pair control scrolling. The left hand key pair can be assigned to a different function.

A function allocation means allows the user to choose which functions are to be allocated to which keys and which mode to use. He may therefore choose functions which have the maximum control facility and which are to have minimal control. Assignment of these keys may be made whilst a performance is actually in progress. Selected functions may be moved to manual control for a part of the performance.

This functionality is assigned as follows. The CPU maintains a map between each key and the function that it will command. The CPU executes the command from the map whenever the associated key is actuated. If the key function is reassigned, the CPU modifies this map so that later actuations of that key cause the different, reassigned function to be executed.

The panel manager software module handles most keypress events in the sub-panel **11**. To control the lamp functions, the panel manager module sends data to an appropriate one of a plurality of function driver modules tailored to the requirements of that function. The panel manager keeps track of which preset keys are assigned to which driver, so that when a particular key is pressed, the panel manager can pass data to the appro-

priate driver. When it is a preset key which has been pressed, the panel manager calculates the preset number based on the current scroll position and makes a call to the appropriate driver module which returns the appropriate preset value. The panel manager retrieves data identifying the currently selected lamp channels from the map which is maintained by the selection manager and makes another call to the driver passing to it the preset number and the data identifying the lamps selected. The driver module uses this data to assemble a message to be passed to the lamps over the network. A copy of the message is also passed to the cue manager so that it can be recorded if required.

Whenever a scroll key is pressed, the panel manager sends an enquiry message to the driver module asking for a name for each key in the new preset list. This allows the names and preset values to be stored in the driver module's private data area—which may be in a disk file or in RAM.

The driver module has a name field which is stored at a predetermined offset from the start address of the driver module. The panel manager can call up this name from the driver and pass it to the dual port memory for display.

In more detail, the following sequence of events occurs. The panel manager repeated makes calls to a "GETKEY" routine to see whether any keypress events are pending. If an event is pending "GETKEY" returns the key number. The panel manager uses this number to look up the key in a key translation table, which contains a key type number and a key value. The key type identifies the key as belonging to a particular functional group of keys, such as "select" keys, "manual" keys and "number" keys. The key value identifies the key within its type. The key type determines which of a number of different routines is next executed. If the key was a "manual" key, the key value passed indicates to the manual handler routine whether it is in either of the two scroll pairs, or in a preset select row. The handler routine uses the key value to select a "manual control record" relating to the appropriate group of keys, which record contains all the information needed to perform the actions required. In particular it contains the number of the driver module(s) associated with each of the three areas of the manual key group.

The handler routine can determine the mode of the group by comparing the driver numbers of the three areas. If the right scroll driver is the same as the preset driver, for instance, the handler recognises that the right scroll keys are scroll keys for the group. If it is either of these that is pressed the handler calculates the index of the next group of presets and calls the driver once per preset key to provide a name for each key to display.

If the key was a "PRESET" key, the handler calculates the index value of the key, and calls the driver to determine the preset value assigned to the key. This allows the driver to maintain its own private storage in appropriate format. The handler then calls the driver again, using the returned value to output the value to the selected lamps. The selection bitmap, the time to execute and the value are supplied in a "stack frame" within the call. The same parameters, plus the driver number, are supplied to the cue manager so that the call can be recorded in the cue list.

When recorded cues are replayed, the cue manager replicates the event directly by interacting with the driver, making the dialog between the handler and the driver unnecessary.

There are many functions which involve changing the state of console parameters during a performance. These may range from executing a cue which is not the next in the numerical sequence to re-arranging the manual driver assignment. The functions are managed by a console driver module which operates as a function allocation means. A scrollable list of function keys and may be sited on any manual group. Each "page" of this driver accesses different functions, thereby allowing cue management, console re-arrangement and other functions to be carried out. This arrangement gives the advantage of enabling the storage of these actions as part of a cue. Manual assignments stored in this way become part of the cue and will be repeated when the cue is replayed. The desired assignments will become part of the performance and will be made available to the operator at the correct point in the execution of the cue list.

Assignment of the potentiometers to master control of cues is also achieved through the console driver.

The various potentiometers in sub-panels **12** and **13** serve the same logical function. Each control group consists of a potentiometer with at least one associated key switch element. The console driver allows the user to allocate the function of any potentiometer to master control of any masterable function—i.e. any function which performs an action which may include an execution time. The master control provides a manually controlled (instead of timed) transition between prior and new values of a selected function.

Most console functions have more than one mode, affecting how the actions selected will be performed. These modes are specific to each driver and are handled on a driver by driver basis. This handling is the responsibility of the panel manager.

When a driver control key on the sub-panel **11** is pressed, the panel manager checks the state of "MODE" key in the set of keys **45** above the numerical key pad **44**. If this key is being held down, the panel manager sends a "set mode" message to the driver, which then conducts a dialog with the user to ascertain which mode is to be established. This dialogue uses the display area **41**.

Editing of driver presets (i.e. changing the values stored) is carried out when a preset select key is pressed while an "EDIT" key in key set **45** is held down. The driver then conducts a dialog with the user to allow editing. Preset values can be entered manually using the numeric keypad, or using one of the encoders **72**.

Two additional patentable aspects of the present invention are also contemplated as supplemental aspects of the above. The first is the macro command. As least some of the keys on the keyboard may be assigned as a macro key. Depressing that particular key commands a plurality of different sub-functions to be executed. This is carried out by associating a function sequence in the map with the key. Subsequent actuations of the macro key command a read and execute of the commands that are previously stored and are associated with that key sequence. The stored commands are executed in the order stored—optionally at times which are also set into the map. For example, the map might appear as follows:

(Key # function parameter function parameter)
<11> <E42> <1FH> <OFF> <01> . . .

indicating that key number **11** executes function E42, parameter 1Fh etc. One function, e.g. OFF, indicates a delay by the amount of the parameter that follows.

This is highly advantageous especially when combined with a allocatable keyboard: since it may be possible to configure the keyboard to eliminate some functions from it entirely. Instead, some functions could only be executed via a macro key. For instance, if function **1** will always be followed by functions **2**, **3**, and **4**, the keyboard need not be programmed to include functions **1**, **2**, **3**, and **4** separately. Instead, during initial setup, these functions are programmed into a single macro key. Only the macro key need later be depressed, and no separate depressions of the separate keys are necessary.

A second advantageous aspect of the present invention is the snapshot feature. The prior art has taught storage of cues, where each cue represents a particular state of a particular light. The inventors of the present invention have found that sometimes a unique lighting effect might be created by the entire system of lights, and that one might want to recreate this overall effect. In the past, it has been necessary to take notes about how this unique effect was determined. According to the present invention, a "snapshot" function is available, to memorize all positions and all parameters of all lights. This is done according to the present invention by making a copy of the contents of the entire block of main RAM, which stores all of the information for all lights. These contents are assigned a snapshot number, which may be executed by a recall snapshot key. A recall of the snapshot memorized in this way allows the entire unique effect to be recalled and displayed at any desired time.

We claim:

1. A stage lighting control console including a panel, a plurality of control elements mounted on said panel and including manually actuable switch elements and manually adjustable control elements and an electronic control system controlled by said switch element and manually adjustable control elements for producing output signals for controlling a multiplicity of independently controllable functions of a plurality of remote lamp units, said electronic control system including function allocation means for determining which of the switch elements and manually adjustable control elements exercise control over each one of the functions of the lamp units, said function allocation means being under the control of a user of the console utilizing selected ones of the switch elements.

2. A stage lighting control console as claimed in claim **1** in which the individual switch elements incorporate in built display devices which are controlled by the electronic control system to provide a display appropriate to the current function of the switch element.

3. A stage lighting control console as claimed in claim **2** in which the panel also includes a plurality of further display areas adjacent individual switch elements or control elements or adjacent groups of such elements to display alphanumeric information indicating the current function of the associated switch or control element or the associated group of such elements.

4. A stage lighting control console as claimed in claim **1** in which the panel comprises a plurality of sub-panels each with a different arrangement of switch elements, control elements and display areas, each sub-panel having its own associated processor unit for controlling the display areas of that sub-panel and the console including a main processor unit which communicates with the sub-panel processor units.

5. A stage lighting control console as claimed in claim **4** in which there is provided a dual-port memory unit which is connected to the main processor unit and to all the sub-panel processor units, such memory unit being used to store switch and control element operation data and display data, the main processor unit reading from the memory unit the switch and control element operation data and writing to the memory unit display data, and the sub-panel processor units writing switch and control element operation data to the memory unit and reading therefrom the display data.

6. A stage lighting control console comprising the combination of a panel, a plurality of user input control elements on said panel, a main processor unit for processing data input by a user to generate a multiplicity of data message blocks to be sent respectively to a multiplicity of individually controllable lamp units, and a distribution unit including a multiplicity of individual serial communication controllers for connection to respective ones of the lamp units, first data bus means connecting said main processor unit to said distributor unit for transferring said message data blocks to said distribution unit and second processor means for distributing the data contained in said message data blocks to said individual serial communication controllers.

7. A stage lighting control console as claimed in claim **6** in which the main processor unit has main RAM in a portion of which it stores said message data blocks, and said distribution unit has a plurality of blocks of dual port RAM into which the contents of said portion of the main RAM can be copied over said first data bus under the control of said second processor means.

8. A stage lighting control console as claimed in claim **7** in which there are a plurality of third processors units for controlling transfer of data from the blocks of dual port RAM to the associated serial communication controllers,each of said third processors being associated with a different one of the blocks of dual port RAM and with a different group of the serial communication controllers.

9. A stage lighting control console as claimed in claim **8** in which each of said third processor units is programmed to transfer data from the associated block of dual port RAM to the serial communication controllers of the associated group in an interleaved byte-by-byte fashion, each serial communication controller transmitting each byte on receipt thereof.

10. A console as in claim **1** wherein said function allocation means includes means for assigning a plurality of different functions to a single actuable switch element, such that actuation of said actuatable switch element causes execution of said plurality of functions in the same order as that in which they were stored.

11. A console as in claim **1** further comprising a snapshot control mechanism, which when actuated causes parameters for a plurality of said multiple lamp units to be stored in a way such that they can be later recalled.

12. A console as in claim **11** wherein said snapshot control mechanism includes means for storing data from all of said lamp units and one of said switch elements

can be allocated by said function allocation means to initiate a recall of any particular snapshot.

13. A stage lighting control system for controlling a plurality of remotely-controllable lamp units, comprising:

a plurality of control elements for commanding control of parameters of said lamps;

a memory area, having a plurality of memory elements, said memory elements storing a value indicative of characteristics of parameters of said lamps;

a processor which receives commands from said plurality of control elements, and responsively alters values in said memory elements based on said commands, and which produces control signals for the lamp units based on the values in the memory area;

a snapshot store control element which, when actuated, commands a snapshot store command whereby all values of all of said memory elements are stored in a memory location other than in said memory element, for later recall; and

a snapshot recall control element which when actuated commands recall of a stored snapshot,

said processor including:

a storage control element which stores said all values of said all of said memory elements in said other memory location responsive to said snapshot store command and which reads out said values responsive to said snapshot recall command.

14. A stage lighting control system as in claim 13, wherein said processor stores a map indicating what specific functions are controlled by each of said control elements, and includes an element for altering said map to allow reconfiguration of said control elements to command different ones of said specific functions, said specific functions including said snapshot store command and said snapshot recall command.

15. A stage lighting control system for controlling a plurality of remotely-controllable lamp units, comprising:

a plurality of control elements for commanding control of parameters of said lamps;

a lighting memory area, having a plurality of memory elements, each of said memory elements having a value indicative of characteristics of parameters of said lamps;

a processor which receives commands from said plurality of control elements, and responsively alters values in said memory elements based on said commands, and which produces control signals based on the values in the memory area; and

a macro recall control element which when actuated commands recall of a stored sequence of commands, and causes execution of said sequence of commands by said processor.

16. A stage lighting control system as in claim 15, further comprising a macro store control element which, when actuated, memorizes a sequence of said commands, and stores said sequence in a memory location other than said lighting memory area, for later recall.

17. A stage lighting control system as in claim 15, wherein said processor stores a map indicating what specific functions are controlled by each of said control elements, and includes an element for altering said map to allow reconfiguration of said control elements to command different ones of said specific functions, said specific functions including said macro store command and said macro recall command.

* * * * *

EXHIBIT B

US005502627A

United States Patent [19]

Hunt et al.

[11] Patent Number: 5,502,627

[45] Date of Patent: Mar. 26, 1996

[54] STAGE LIGHTING LAMP UNIT AND STAGE LIGHTING SYSTEM INCLUDING SUCH UNIT

[75] Inventors: Mark A. Hunt, Derby; Keith J. Owen, Moseley; Michael D. Hughes, Wolverhampton, all of United Kingdom

[73] Assignee: Light & Sound Design Limited, Edinburgh, United Kingdom

[21] Appl. No.: 77,877

[22] Filed: Jun. 18, 1993

[30] Foreign Application Priority Data

Sep. 25, 1992	[GB]	United Kingdom	9220303
Sep. 25, 1992	[GB]	United Kingdom	9220309
Apr. 20, 1993	[GB]	United Kingdom	9308071

[51] Int. Cl.^{6} F21V 21/00

[52] U.S. Cl. 362/286; 362/233; 362/386; 318/603

[58] Field of Search 362/233, 285, 362/286, 276, 386, 293, 319; 318/574, 601, 603

[56] References Cited

U.S. PATENT DOCUMENTS

3,731,177	5/1973	Commander et al.	318/603
5,045,983	9/1991	Shields	362/293

FOREIGN PATENT DOCUMENTS

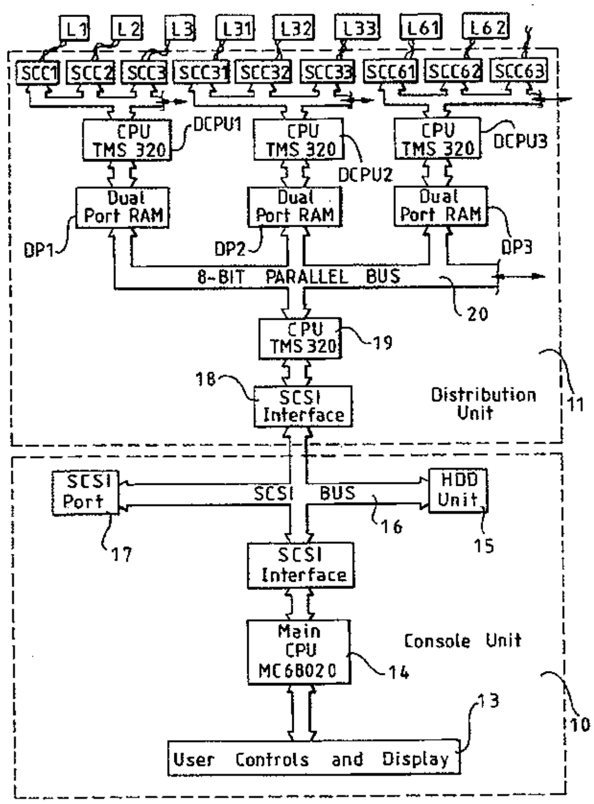
0495305	7/1992	European Pat. Off. .
2643134	8/1990	France .
2209229	5/1989	United Kingdom .
2231138	11/1990	United Kingdom .
8905421	6/1989	WIPO .

Primary Examiner—Stephen F. Husar
Attorney, Agent, or Firm—Fish & Richardson

[57] ABSTRACT

A stage lighting lamp unit includes a processor for receiving control data from a remote console. Beam orientation data for the lamp unit is passed to the lamp in the form of the x, y and z co-ordinates of a point in space through which the beam is to pass. The processor divides the required lamp travel into a number of stages dependent on execution duration data sent with the position data, and calculates, for each stage, a new value for pan and tilt angles for the lamp. These values are passed to pan and tilt controlling co-processors which control servo-motors for pan and tilt operation. The lamp unit also incorporates a rotatable shutter for interrupting the lamp beam when required. The shutters of all the lamps in a system can be instructed from the remote console to open and close in synchronism, thereby providing a stroboscopic effect.

9 Claims, 10 Drawing Sheets



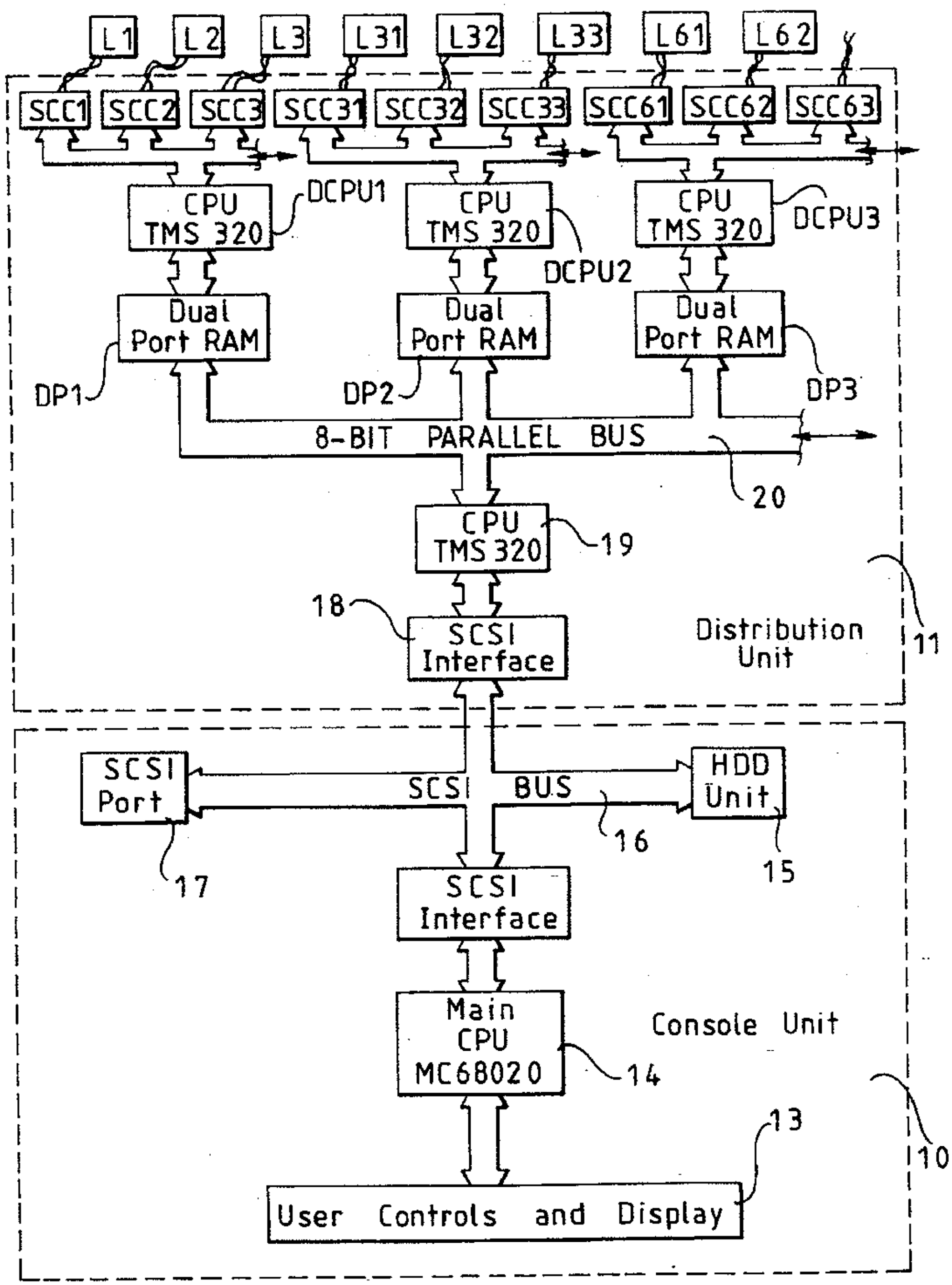
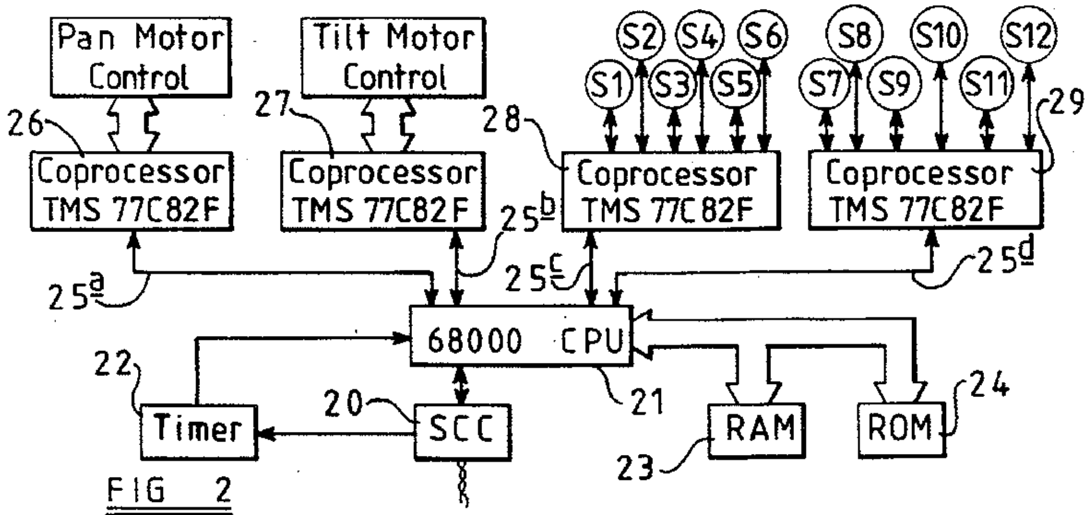
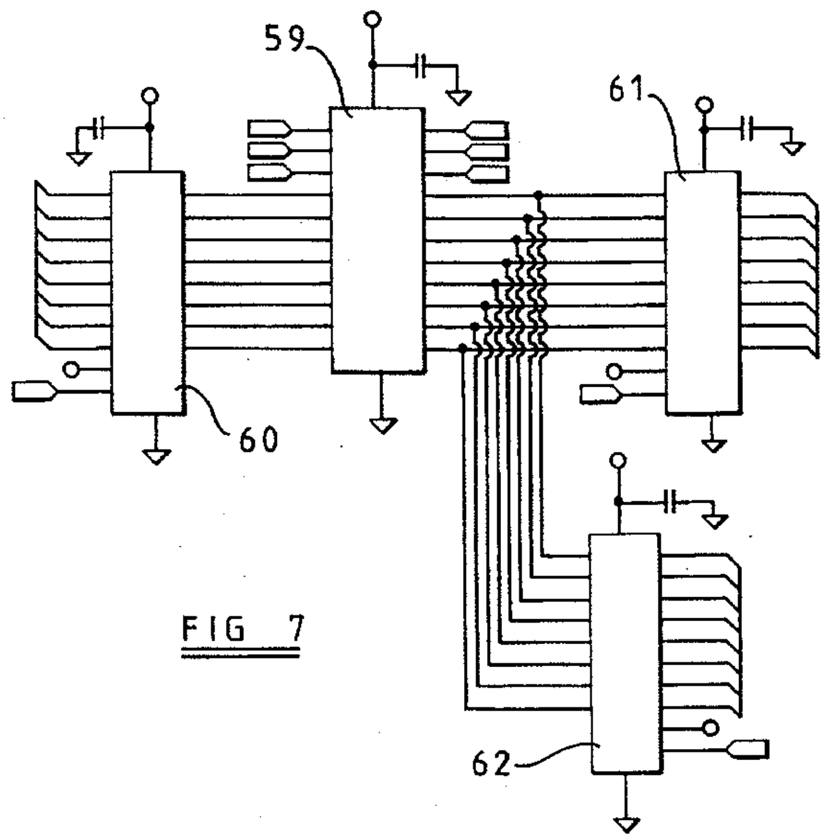

L1
L2
L3
L31
L32
L33
L61
L62
SCC1
SCC2
SCC3
SCC31
SCC32
SCC33
SCC61
SCC62
SCC63
CPU TMS 320
DCPU1
CPU TMS 320
DCPU2
CPU TMS 320
DCPU3
Dual Port RAM
DP1
Dual Port RAM
DP2
Dual Port RAM
DP3
8-BIT PARALLEL BUS
20
CPU TMS 320
19
18
SCSI Interface
Distribution Unit
11
SCSI Port
SCSI BUS
HDD Unit
16
15
17
SCSI Interface
Main CPU MC68020
Console Unit
14
13
10
User Controls and Display

FIG 1



Pan Motor Control
Tilt Motor Control
26
27
28
29
Coprocessor TMS 77C82F
Coprocessor TMS 77C82F
Coprocessor TMS 77C82F
Coprocessor TMS 77C82F
S1
S2
S3
S4
S5
S6
S7
S8
S9
S10
S11
S12
25a
25b
25c
25d
68000 CPU
22
20
21
24
Timer
SCC
RAM
ROM
23

FIG 2



59
61
60
62

FIG 7

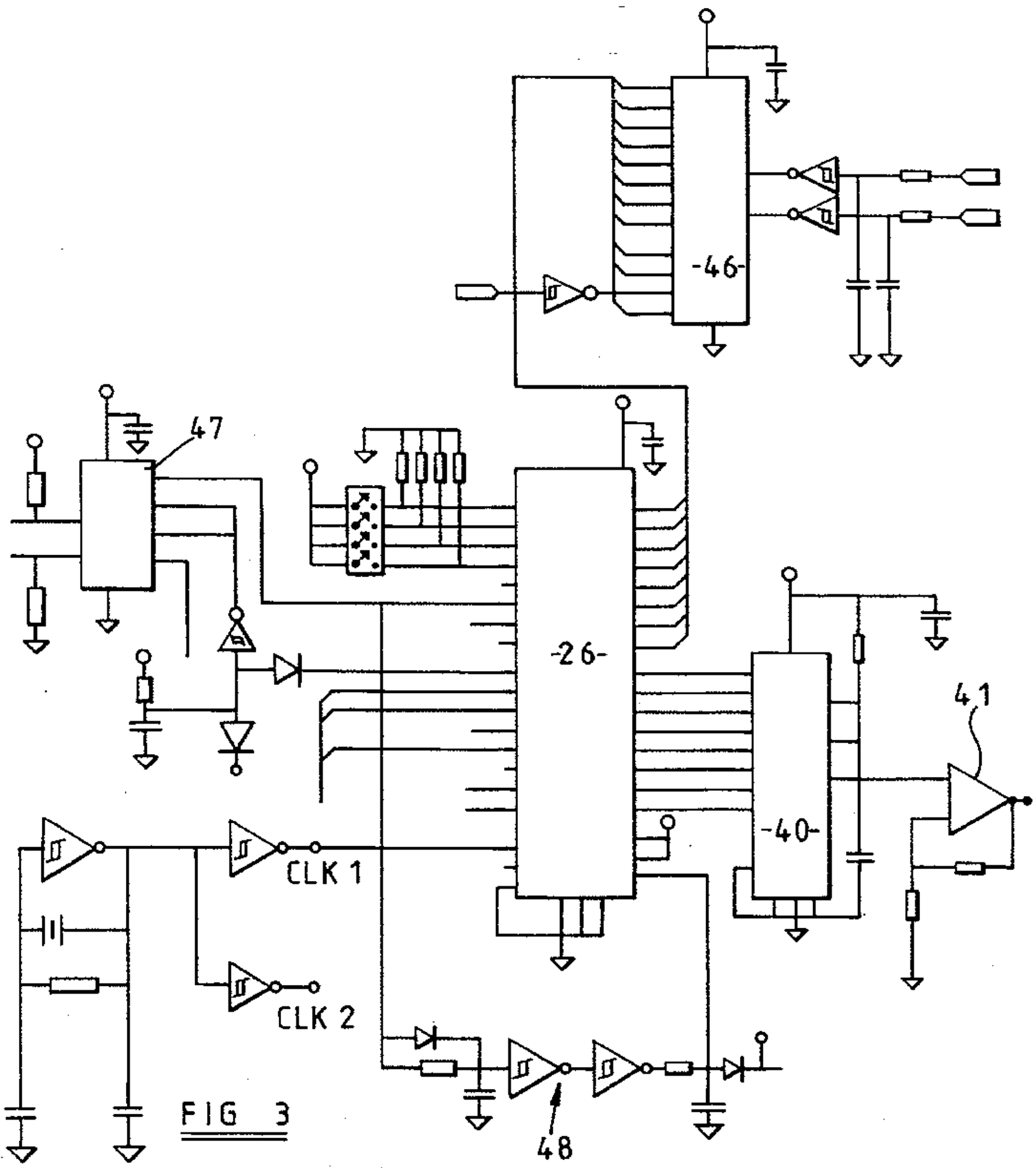


FIG 3

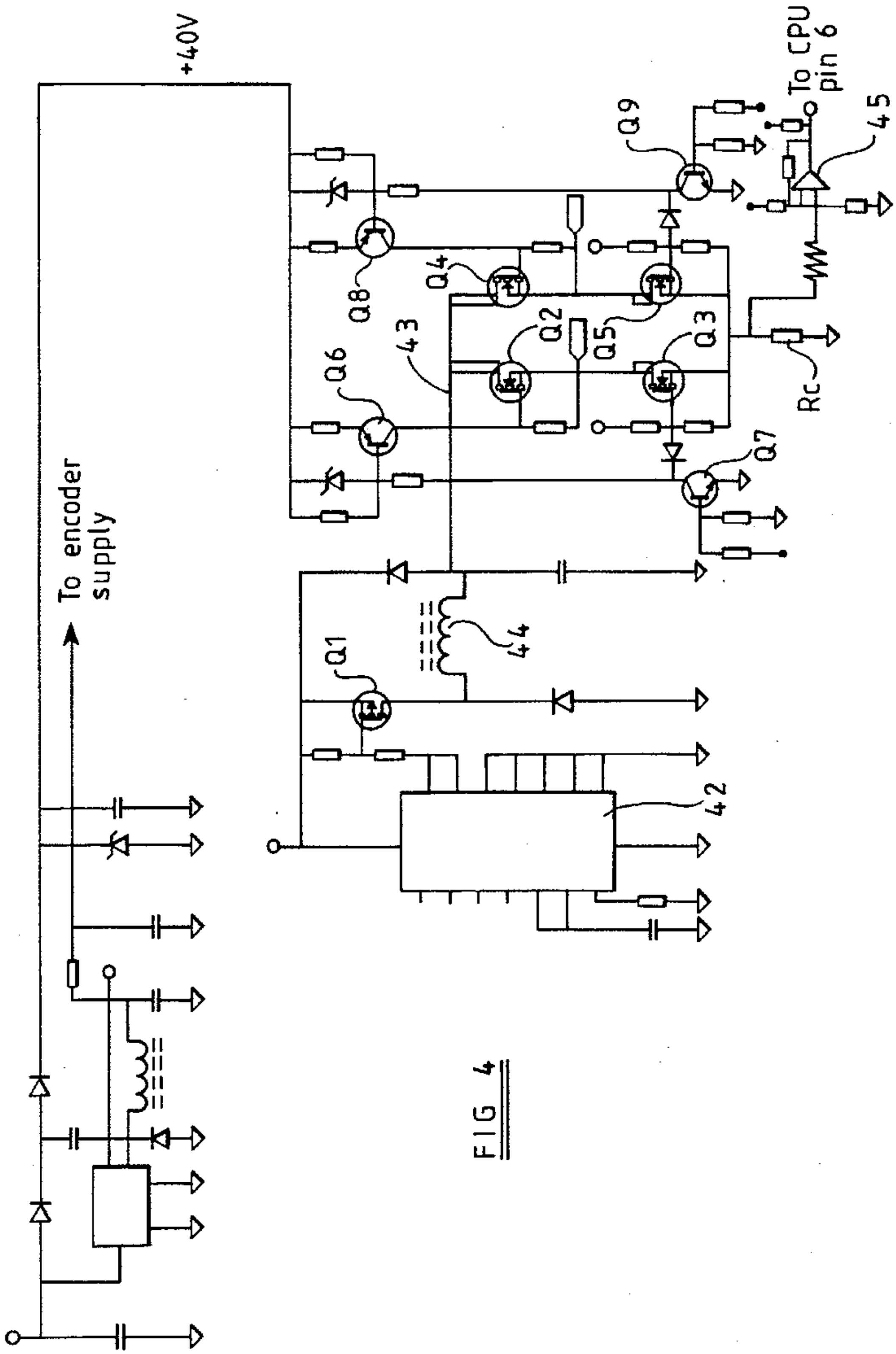

+40V
To CPU pin 6
45
Q9
Q8
Q4
Q2
Q5
Q3
43
Q6
Rc
Q7
To encoder supply
Q1
44
42

FIG 4

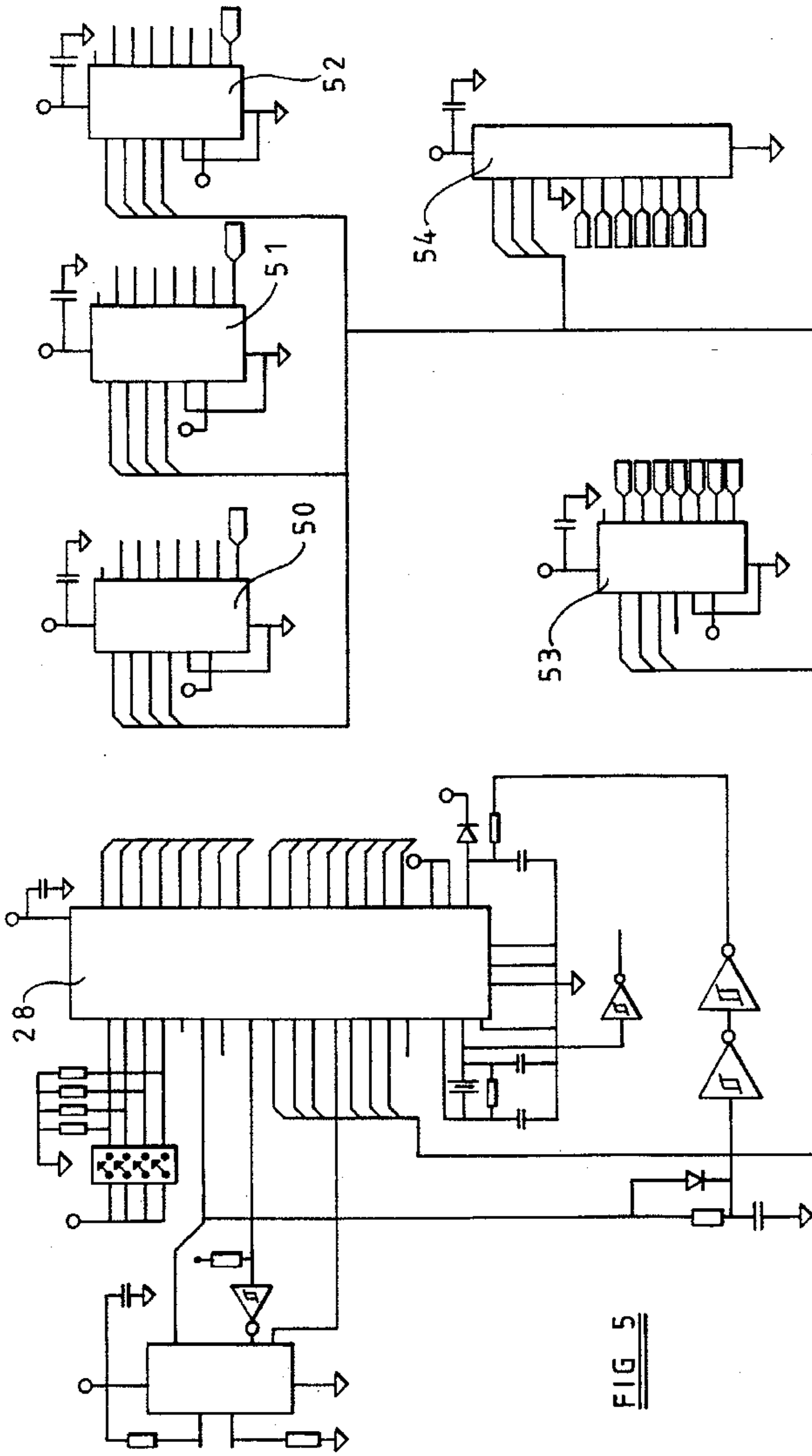


FIG 5

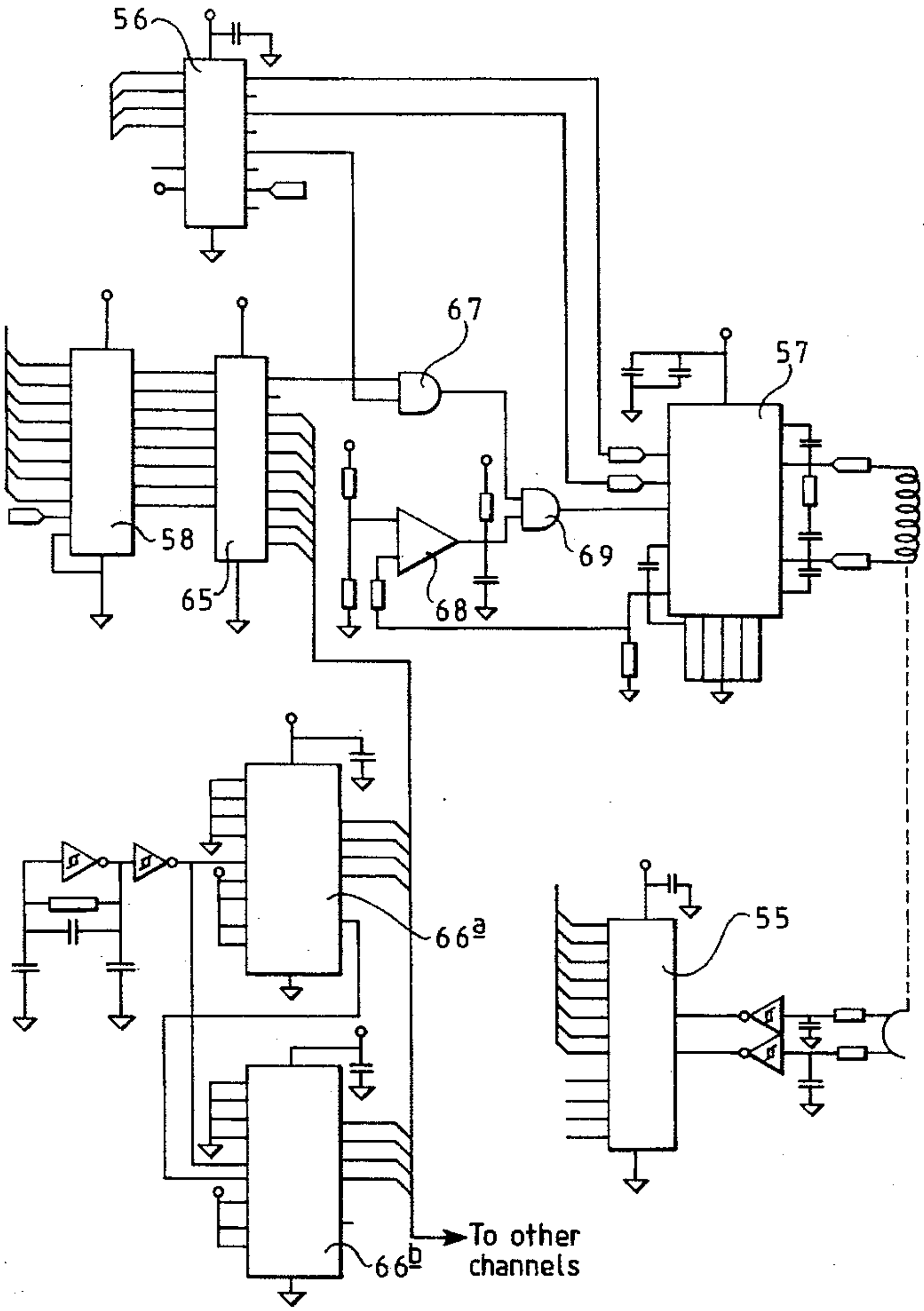
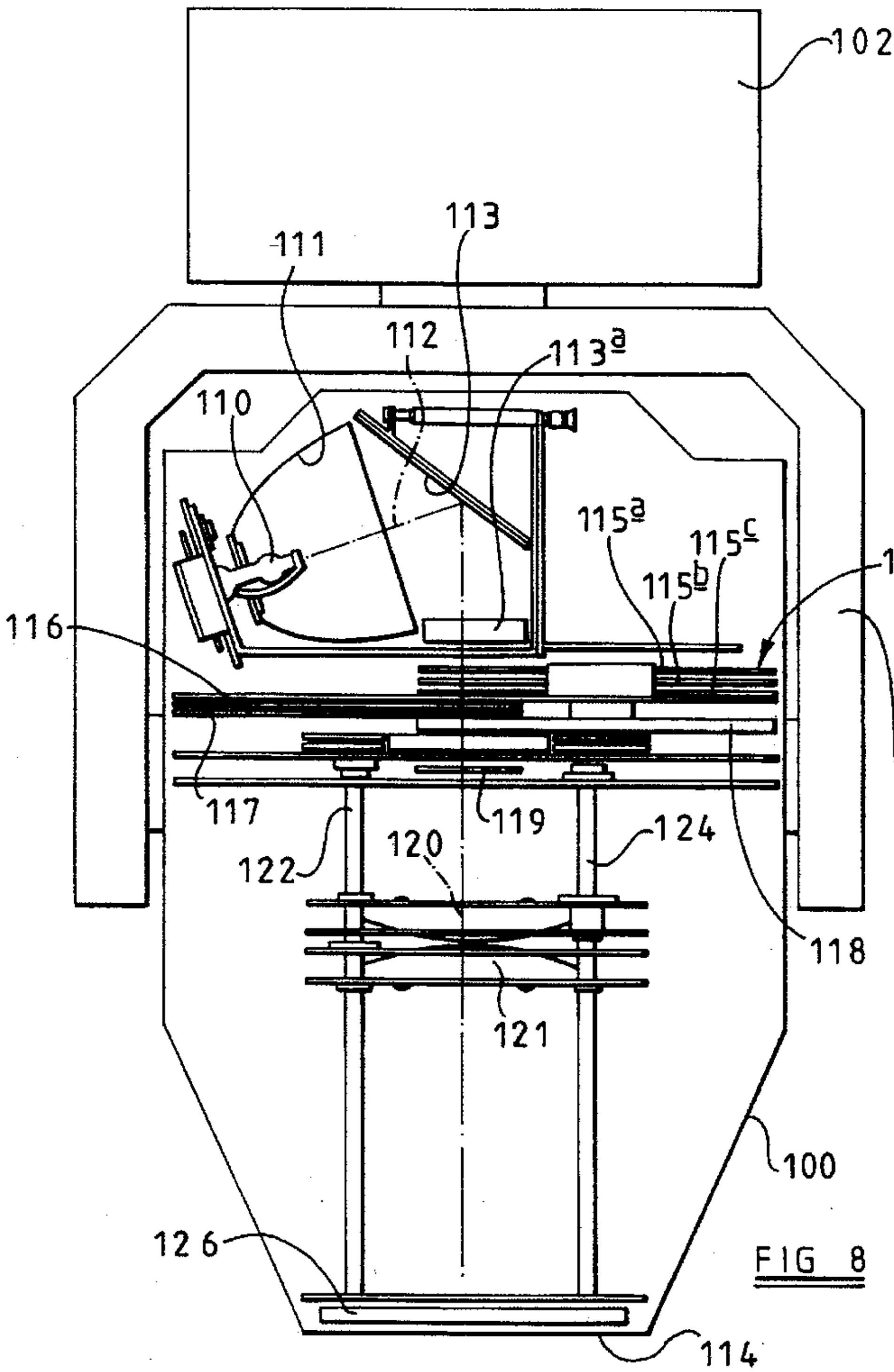


FIG 6



102
111
113
112
113a
110
115a
115c
115
115b
116
101
117
119
124
122
120
118
121
100
126
114

FIG 8

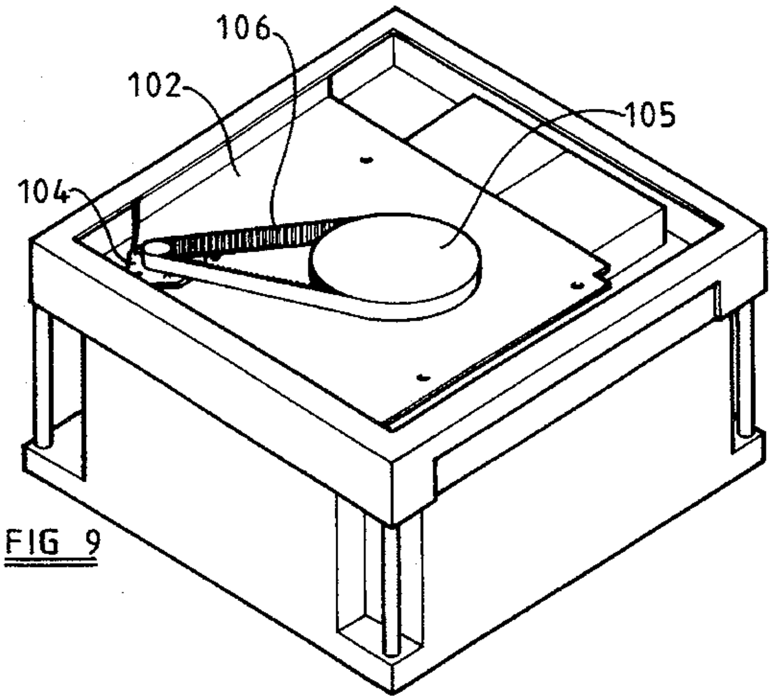


FIG 9

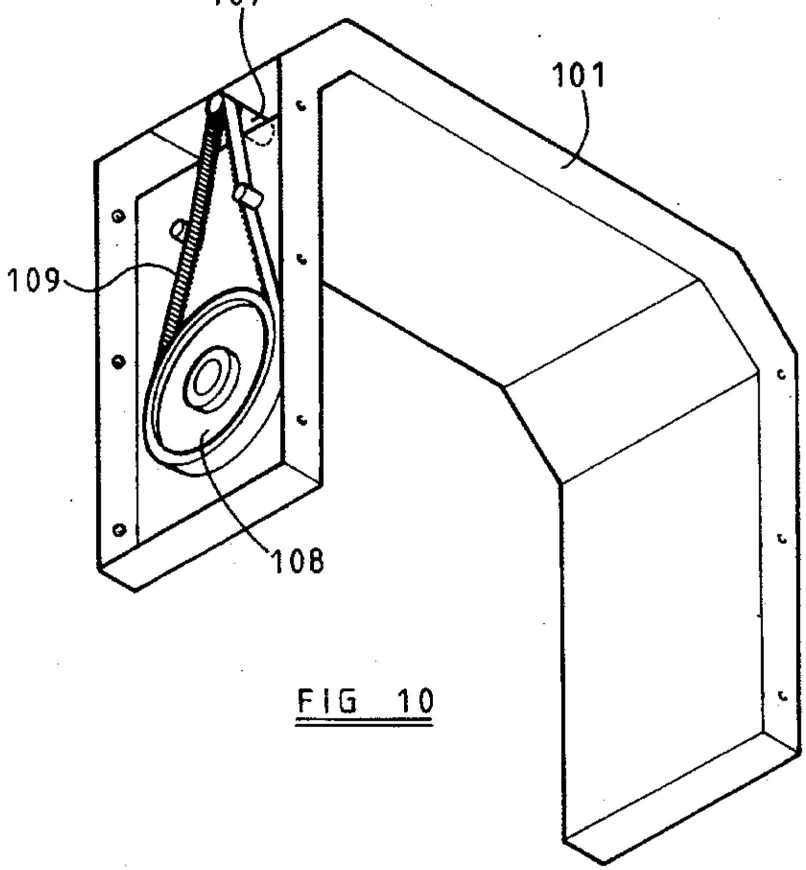


FIG 10

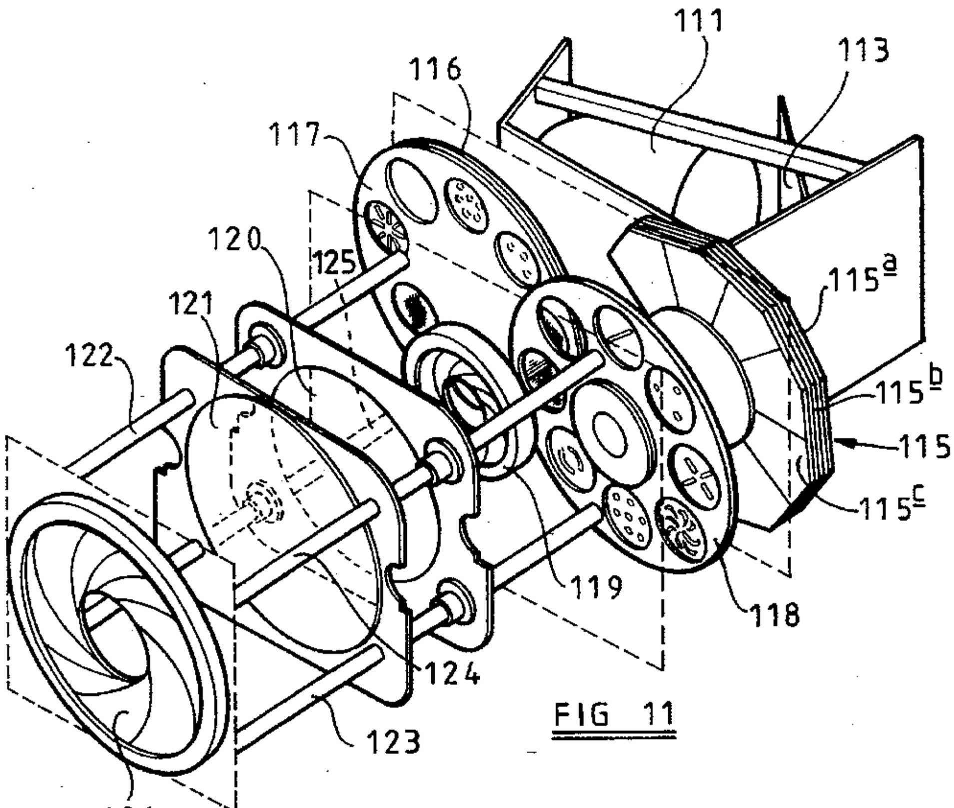

111
113
116
117
120
125
121
122
115a
115b
115
115c
119
118
124
123
126

FIG 11

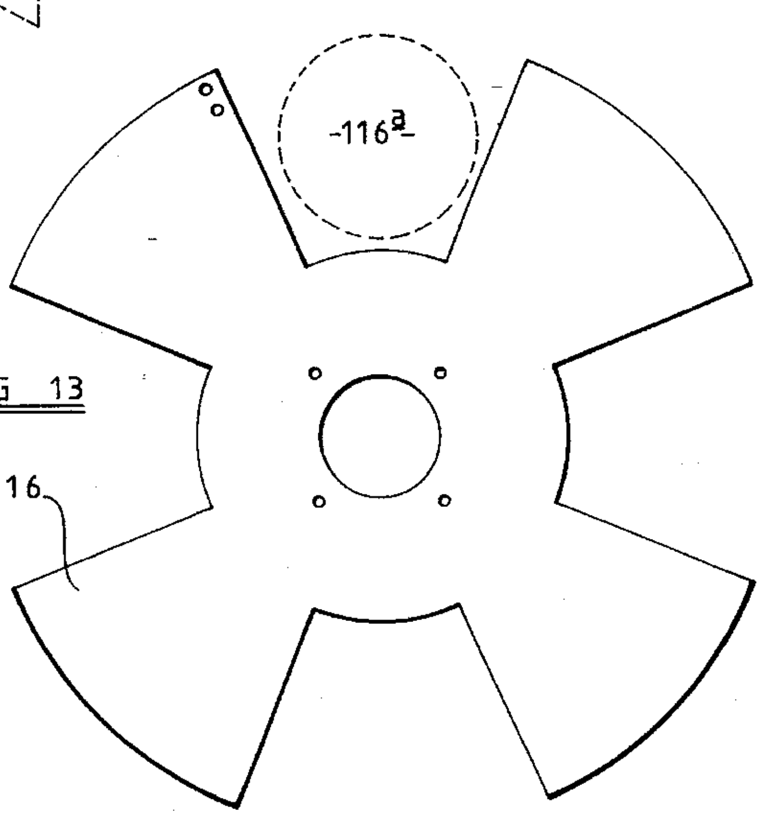

116a
116

FIG 13

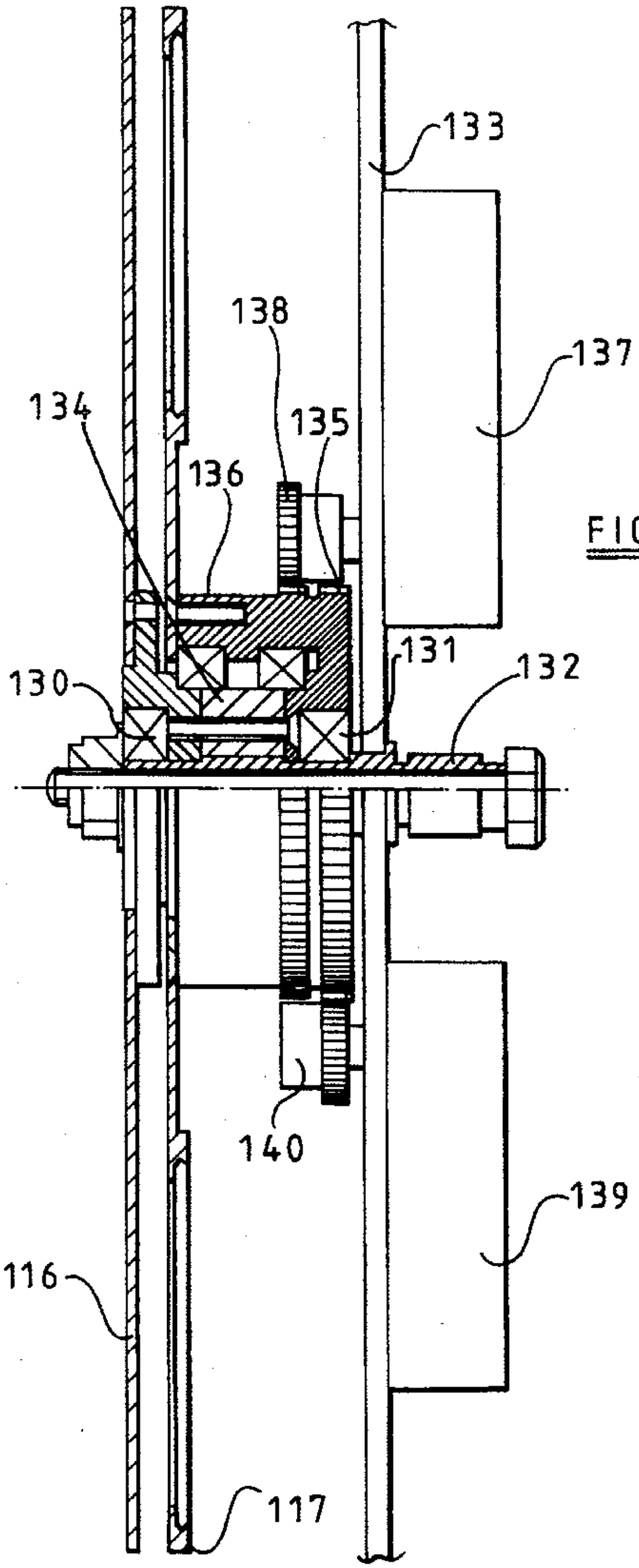


FIG 12

STAGE LIGHTING LAMP UNIT AND STAGE LIGHTING SYSTEM INCLUDING SUCH UNIT

This invention relates to stage lighting and is particularly concerned with the control of multiple functions of a lamp.

It has already been proposed to incorporate in a lamp unit a plurality of different functions, such as colour changers, focusing lenses, iris diaphragms, gobo selectors and pan and tilt mechanisms which are controlled from a remote console. Stage lighting systems have as a result reached very high levels of complexity requiring a very complicated main control console and lamp unit constructions. The use of microprocessors, both in the console and the lamps has become conventional as increasing complexity makes it more difficult to produce and subsequently maintain a system which uses hard wired logic or analog controls. In such systems the microprocessor in the console is used to allow the user to set up lighting cues and to control the sending of appropriate data to the lamp microprocessors. The lamp microprocessors are also involved in controlling communication between the console and the lamps, and also have to control a plurality of servomotors which drive the various functions of the lamps.

It is one object of the present invention to provide a lamp microprocessor and servo-control arrangement which allows complex functions to be carried out.

It is another object of the invention to provide a lamp control system in which control of pan and tilt movements of each lamp can be carried out in rapid and efficient manner, enabling large groups of lamps to make co-ordinated movements.

It is yet another object of the invention to provide each lamp in a stage lighting system with a means for quickly interrupting its light beam and quickly re-establishing the beam so that a group of lamps can be made, when required to flash in synchronism.

In accordance with one aspect of the invention there is provided a lamp unit for connection to a remote control console for the control of a plurality of different functions of the lamp, said unit comprising a main processor circuit, associated with a communication controller for accepting message data from the console, a plurality of servo-controls for operating said functions of the lamp, and a plurality of co-processors which are connected to the main processor circuit so as to be supplied thereby with desired value data for the various lamp functions, said servo-controls being controlled by said co-processors.

In the case of pan and tilt controls where close control is required throughout the movement of the lamp from an initial position to a new position, one of the co-processors is assigned solely to the control of movement about each axis. Other functions can share a co-processor.

The main processor circuit of the lamp is preferably programmed to accept data from the control console defining not only a target position for any function, but also a duration over which the function is to be executed. In this case the main processor circuit divides the "journey" into segments and updates the target position data passed to the associated co-processor at intervals.

In accordance with another aspect of the invention, there is provided a lighting control apparatus comprising the combination of a main control console for accepting user input relating to required beam movements, a plurality of independently operable lamp units situated remotely from the console, each of the lamp units incorporating a servo-mechanism for automatically moving the lamp beam about two mutually transverse axes to a desired angular position and data communication means connecting the console to the lamp units for the transmission of desired position data to the lamp units, the desired position data being transmitted in the form of a set of three dimensional linear co-ordinates defining a point in space through which the lamp beam is required to pass, and each lamp unit including a calculating device for calculating the desired angular position from the desired position data and supplying the servo-mechanism with such desired angular position.

In addition to the "point at" mode of operation mentioned above, additional modes may be specified in which the lamps point away from the specified point or in which they all point in the same direction parallel to a line between a fixed position in the co-ordinate system and the specified point.

Conveniently, all the data concerning the positions and orientations of the individual lamp units within the co-ordinate system is stored in a setup file kept on a hard disk drive in the console. When the same lighting set-up is used at different venues, where it is impossible to set the frame which carries all the lamp units at exactly the same position as that for which the set-up was designed, offset data can be input at the console and either used within the console microcomputer to correct the position data stored during set-up as it is sent out, or such data can be sent to all of the lamp units over the network and stored there, to enable the corrections to be made in the individual lamp processor units.

In accordance with another aspect of the invention, a stage lighting unit comprises a housing, a light source within said housing, an optical system for forming light from said light source into a beam, a rotary shutter device having a plurality of blades, said shutter device being rotatably mounted in the housing so as to cause said blades to pass through and obstruct said beam as the shutter device rotates, a motor for rotating said shutter device and a servo-control for controlling said motor in accordance with data received in use from a remote control console.

The invention also resides in a stage lighting system incorporating a plurality of lighting units as defined above controlled by a common remote control console via data communication means, whereby the rotary shutter devices of all the units can operate in synchronism.

An example of the invention will now be described with reference to the accompanying drawings, in which:

FIG. **1** is a block diagram of a stage lighting system;

FIG. **2** is a block diagram of the internal circuitry of one of a plurality of lamp units in the system of FIG. **1**;

FIGS. **3** and **4** are more detailed circuit diagrams showing a pan motor drive control forming part of the internal circuitry of the lamp;

FIGS. **4** to **7** are detailed circuit diagrams showing a rotary shutter motor drive control forming part of the internal circuitry of the lamp;

FIG. **8** is a diagrammatic, part-sectional view of one of the lamps;

FIG. **9** is a perspective view of a pan movement drive arrangement;

FIG. **10** is a perspective view of a tilt movement drive arrangement;

FIG. **11** is a diagrammatic perspective view of the internal moving parts of the lamp;

FIG. **12** is a sectional view showing the drive arrangement for a shutter and a gobo wheel forming part of the lamp; and

FIG. **13** is an elevation of a shutter wheel forming part of the lamp.

Referring firstly to FIG. **1**, the system consists basically of a console unit **10**, a signal distribution unit **11** and a plurality of lamps L1, L2, L3 . . . , L31, L32, L33 . . . , L61, L62 . . . individually connected by twisted pair data communication links to the distribution unit.

The console unit **10** has an array of switches, slider potentiometers, rotary digital encoders and other user actuable input devices (not shown) and a display indicated at **13**. These are all connected to main console cpu **14** (an MC68020 micro-processor) which has the task of receiving inputs from the user actuable input devices and controlling the display. Both tasks are assisted by separate co-processors which directly interface with different parts of the console.

The main cpu can communicate with a hard disk drive unit **15** via a SCSI bus **16** which also connects it to the distribution unit and to an external SCSI port **17**, through the intermediary of which the console can, if required be connected to a personal computer. The user controls can be used in setting up a sequence of cues in advance of a performance, the sequence being stored in a cue file on the hard disk drive unit **15**. The sequence can be recalled during the performance to enable the various stored cues to be executed. Direct manual control of the lamps from the console is also possible as is manual editing of cues called up from the hard disk. The main console cpu **14** creates messages to be sent to the individual lamps, each message comprising a fixed number of bytes for each lamp. The messages contain data relating to the required lamp orientation, beam coloration, iris diaphragm diameter, gobo selection and rotation, zoom projection lens control and opening or closing of a shutter included in the lamp. A block of the RAM of the main cpu is set aside for the storage of these messages, the block being large enough to contain messages for **240** lamps, being the largest number which can be controlled via the distribution unit. Where it is required to control more than **240** lamps additional distribution units can be connected to the SCSI bus and extra main cpu RAM reserved for message storage. When any message data is changed the main cpu **14** sets a flag in the RAM block which is detected at a given point in the main cpu program loop and interpreted as a signal that the changed message data is to be transferred to the distribution unit **11**.

The distribution unit **11** has a main cpu **19** which controls reception of data from the SCSI bus interface and distribution of such data to up to eight blocks of dual,port memory DP1, DP2, DP3 . . . via an eight bit data bus **20**. The cpu **19** is alerted to the waiting message data when cpu **14** selects the distribution unit. The cpu **19** then supervises byte by byte transfer of the message data which it routes to the various blocks of dual port memory.

For actually sending out the message data to the lamps, there are a plurality of serial communication controllers SCC1 to SCC30, SCC31 to SCC60 etc, there being thirty serial communication controllers associated with each block of dual port memory. A further cpu DCPU1, DCPU2, etc is associated with each block of dual port memory and distributes message data transferred to the dual port memory to the individual serial communication controllers and the messages are transferred to the lamps. Each serial communication controller in the distribution unit includes a line driver which can be disabled except when data is to be transmitted. Enabling of the driver can cause a spurious signal to be transmitted over the data link. To allow such spurious signals to be identified and ignored, a two-byte gap is left between enabling the line driver and commencing transmission of the message data for the channel in question.

This will be described in more detail herein. All asynchronous serial communication systems require framing information to synchronize the reception process. This has been typically done in the prior art using start bits and stop bits.

The present invention preferably uses FM0 coding in which the data is transmitted as one cycle of the carrier frequency for a zero or as a half cycle of the carrier frequency for a one. When the line has been idle, no waveform at all is present. When the line drivers are first enabled, an arbitrarily short pulse will usually appear on the line, due to lack of synchronization between the data signal and the enabling signal. This short data pulse could be misinterpreted as a start bit, for example and if so it would disturb later framing.

The present invention avoids any problems from this arbitrarily short pulse. To avoid this, the present invention uses a timer on the receive line, set to the time needed to receive two bytes on the serial data line. This timer is restarted whenever a byte on the data line is detected.

Each time the timer interrupt occurs, the number of bytes received is checked against the number of bytes in a valid data frame. If the number is incorrect, then the count is cleared and the message is discarded. If correct, the information is passed to the main program loop by setting a flag variable.

When the data line is first enabled, the distribution box has an internal delay of at least two byte times, which must elapse before any data will be sent. Any data received by the lamp will therefore be discarded as noise by the timer interrupt routine. After that, the real data can be safely sent down the line since the start bit of the first byte will be received correctly. When the transmission is completed, the line drivers will be disabled again.

Each of the cpus eg DCPU **1**, transfers data from the associated dual port RAM DP1 to the serial communication controller SCC1 to SCC30 with which it is associated one byte at a time, ie the first byte for SCC1 is transferred followed by the first byte for SCC2 and so on, each serial communication controller commencing transmission as soon as it has received its byte of data. The serial communication controllers operate to transmit data at 230.4 Kbps so that it takes about 35µs to transmit each byte. Transfer of data from the dual port RAM DP1 to the serial communication controllers is, however, at a rate of several Mbps, so that the transmissions from all the serial communication controllers are almost simultaneous. The cpu DCPU **1** is not required to monitor the transmission of data by the serial communication controller, but utilizes a software timer to commence transfer of the second byte to the serial communication controllers. This timer is started when transfer of the byte of data to the last serial communication controller SCC30 has been completed and its time-out duration is slightly longer than the byte transmission time, say 40µs. Transmission of all the messages takes about 1.5 ms out of a distribution unit main program loop duration of 4 ms.

As shown in FIG. **2**, each lamp includes a serial communication controller **20** which controls reception of message data from the individual data link connecting it to the distribution unit **11**. The receipt of any signal from the data link causes an interrupt of the lamp main cpu **21** (another MC68000) and the cpu **21** then controls acceptance of the signals. A timer 22 times the gaps between bytes received from the data link and this timer causes another interrupt on time-out. The time-out time of the timer is between the times taken to transmit 1 and 2 bytes, so that time out always occurs following a spurious signal caused by line driver

enabling. The time-out interrupt causes the cpu **21** to inspect the total number of bytes received since the initial interrupt and if this is less than the expected number of bytes (which is constant) the message is ignored. The time-out interrupt also resets a software data pointer to the beginning of a receive buffer in readiness for the next transmission.

The cpu **21** operates in accordance with programs stored in the lamp cpu ROM. On receipt of a message of valid length, a program variable representing the number of messages received since the lamp program was last started is incremented and the main program loop of the lamp cpu checks this variable every 16 mS. If the variable has changed since the last check, the data in the receive buffer is compared with corresponding values of variables representing current "desired values" of the various lamp function parameters. For example the receive buffer may contain two bytes representing the x, y and z co-ordinates of a point in an orthogonal three dimensional frame of reference, through which point it is required that the axis of the lamp beam should be directed. If the values of the corresponding byte pairs in the receive buffer and the desired value variables already contained in the cpu RAM are the same, no action is taken in respect of the control of the motors which control pan and tilt action of the lamp (to be described in more detail hereinafter).

As shown in FIG. **2**, the main lamp cpu **21** communicates via serial data links **25***a*, **25***b*, **25***c* and **25***d* with four servo-control co-processors **26**, **27**, **28** and **29**. Each of these co-processors is a TMS77C82 cpu. Co-processors **26** and **27** respectively control pan and tilt operation, and each of the co-processors **28** and **29** can control up to six different dc servo-motors operating different functions of the lamp.

Before proceeding with a more detailed description of the circuitry and operation of the lamp electronics, some detail will be given of the various functions of the lamp. FIG. **8** shows the relative positions of a plurality of independently operable beam characteristic control elements within the lamp housing **100**. The lamp housing is pivotally mounted on a U-bracket **101**, which is itself pivotally mounted on a mounting base **102**. FIG. **9** shows the mounting base **102** which incorporates a pan drive motor/gearbox/optical encoder arrangement **104** which drives a gear **105** attached to the U-bracket via a reduction toothed belt drive **106**. FIG. **10** shows how, within the hollow structure of the U-bracket **101**, there is mounted a tilt drive motor/gearbox/optical encoder **107** which drives a gear **108** attached to the lamp housing via another reduction toothed belt drive **109**.

As shown in FIGS. **8** and **11**, within the lamp housing, a light source **110** is mounted within an ellipsoidal reflector **111** providing a light beam with an axis **112** which is reflected by a mirror **113**, which is a dichroic mirror that reflects only visible light and passes ultra voilet and infra red light, the reflected light passing out through an opening **114** at the opposite end of the housing. The reflector **111** has a generally cup-shape surrounding the bulb **110**. According to one aspect of the invention, the axis **112** has an angle pointing in a direction rearward relative to a perpendicular to the central axis **120** of the lamp unit. If the reflector is located as shown, such that an outside edge of the reflector is generally parallel to a rear end of the housing, the optimal packing efficiency is achieved. As shown in FIG. **8**, this allows the reflector to be most efficiently packed into the available space. The reflected beam from the mirror **113** passes firstly through a collimating lens **113***a*, and then the colour changer **115** which comprises dichroic filters having differing transmission characteristics mounted on co-centered three filter disks **115***a*, **115***b* and **115***c* rotable around a

common axis of rotation. Each disk has nine different filters on it and one blank space around its periphery, so that up to 1000 different combinations of filters can be positioned across the beam by selective positioning of the three disks (although not all of these combinations are necessarily useful as some may block all visible light). The blank space of each of the disks can be used to eliminate any color changing characteristic of that disk. These disks are driven by three of the dc servo-motors. Next the light beam passes through the plane of a bladed shutter **116** (shown in FIG. **13**) and a first gobo wheel **117** which has various gobos mounted in or over circular holes therein. As shown in FIG. **12** described in more detail hereinafter, two motors are committed to driving the shutter **116** and the gobo wheel **117** respectively. Next, there is a second gobo wheel **118** on which there are mounted a plurality of gobos which are rotatable relative to the wheel **118**. There is one motor (not shown) for driving the gobo wheel **118** and another for rotating the gobos mounted thereon through a gear arrangement (not shown). Next along the light beam is a beam size controlling iris diaphragm **119** driven by another motor (not shown). Two further motors (not shown) drive two lens elements **120**, **121** along guides **122**, **123** parallel to the beam axis using lead screws **124**, **125**. The lens elements form a simple two element zoom lens controlling the spread and focus of the beam. Finally, an outer iris diaphragm **126** is provided adjacent the opening **114** and this is driven by a further motor (not shown). In the example described, therefore only eleven channels are actually employed.

Referring now to FIG. **12**, the shutter **116** is rotatably mounted on bearings **130**, **131** on a shaft **132** fixed to a mounting panel **133** which is secured to the housing. The gobo wheel **117** is rotatably mounted on bearings on a tubular shaft **134** which acts to space the shutter **116** from a first drive gear **135**. The gobo wheel **117** is actually mounted on a second drive gear **136**. The shutter motor **137** (which is combined with a reduction gearbox and an optical encoder) is mounted on the panel **133** and drives a pinion **138** meshed with the first gear **136**. Similarly motor **139** drives a pinion **140** meshed with the second gear **136**. The shutter has four blades arranged symmetrically around its axis, with the blades and the gaps between them each subtending 45 degrees at the axis. The blades and the gaps between them are wide enough to block or clear the entire cross-section of the beam, shown in FIG. **13** at **116***a*.

Turning now to FIGS. **3** and **4**, the co-processor **26** is shown providing an eight bit data output to a d/a converter **40** (FIG. **3**) the output of which is amplified by an operational amplifier **41** and supplied to the "COMPEN" terminal of an LM3524 pulse width modulator ic **42** (FIG. **4**). The ic **42** control a P-channel enhancement mode MOSFET Q1 which, when switched on, connects a 24V supply to a motor supply bus **43** through the intermediary of an inductor **44**. The motor is connected in a bridge formed by two push-pull pairs of MOSFETs Q2, Q3 and Q4, Q5. These four MOSFETs are driven by respective driver transistors Q6, Q7, Q8 and Q9. Transistors Q7 and Q9 are respectively controlled by "LEFT" and "RIGHT" outputs taken from the co-processor **26**, so that FETs Q2 and Q5 or FETs Q3 and Q4 are biassed to conduct. Transistors Q6 and Q8 are driven from a 40 V supply rail so as to ensure that FETs Q2 and Q4 are turned hard on when conductive, thereby ensuring minimum power dissipation in these devices.

The two FETs Q3 and Q4 are connected to the return bus via a current sensing resister RC, which supplies a current related signal to a voltage comparator **45** with hysteresis to provide an input to the A6 input terminal of the co-processor **26** when the current exceeds a predetermined limit. This enables the co-processor to reduce the power applied to the motor to maintain it within safe operating limits.

The optical encoder of the pan motor provides two digital outputs in quadrature, these outputs being cleaned up by interface circuits and applied to two inputs of an HCTL-2016 counter ic **46** intended specifically for use with quadrature type encoders. The counter **46** counts up when the pulses are in one relative phase relationship and down when the opposite phase relationship exists. It therefore maintains a count-state related to the motor shaft position and hence the pan angle of the lamp. This count-state is applied to the CO to C7 terminals of the co-processor **26**. The co-processor **26** also receives "desired value" data from the main lamp cpu **21**, via a 75176 ic **47** (which in fact serves both co-processors **26** and **27**). The ic **47** is used to control the transmission of data between the main lamp cpu and the co-processors. Normally the ic **47** is set to receive data from the cpu **21** and pass it to the two co-processors **26** and **27**. At power-up or when the main lamp cpu **21** transmits a "break" command, the co-processor **26** is reset by a circuit **48**. The co-processor **26** has a cycle time of 1 mS and on receipt of new data it determines the distance to be travelled and then increases the "desired position" value which is compared with the actual position count by one sixteenth of the required change on each successive iteration of its control loop.

The desired value signals passed from the cpu **21** to the co-processor **26** are also time-sliced, being incremented every 16 mS. When new position data is transmitted to the lamp it is accompanied by data representing the length of time over which the movement is to be spread. The data is received, as mentioned above, in the form of two byte numbers respectively representing the x, y and z co-ordinates of a point in a Cartesian co-ordinate system. During initial setting up of the system, each lamp is sent data which informs its cpu **21** of its position in the coordinate system and also of its orientation.

On receipt of a new set of "point at" co-ordinates, the cpu **21** undertakes a "time-slicing" operation to determine how data should be passed to the co-processors **26** and **27**. First of all, it determines how many 16 mS loops will take place in the time duration determined by the data contained in the message received by the lamp and sets up a variable U equal to the reciprocal of this number. A travel variable P is initialised to zero and the total distance to be travelled is determined for each of the pan and tilt movements. Thereafter, on every iteration of the **16**mS loop the travel variable P is incremented by the reciprocal variable U, the result is multiplied by the total travel required and this is added to (or subtracted from) the previous desired value before transmission to the co-processor **26** or **27**. When the variable P exceeds unity, the target has been reached.

The message sent to the lamp may include a flag indicating whether travel is to occur in a linear fashion as described above or have a sinusoidal profile imposed on it. In the latter case the value of P is modified as follows:

$P'=sin(2*P)+0.5*(P>0.5)$ the latter term being 0 or 1

The main cpu **26** must next convert the x,y,z values into pan and tilt value data for passing to the co-processors **26** and **27**. The cpu first carries out a linear transformation of the absolute x,y,z co-ordinates into co-ordinates x', y', z' relative to the lamp's own frame of reference using the data supplied during initial set up. The ratio of the transformed x' and y' values is calculated as a 16-bit integer, which is used as an index to an ARCTAN table stored in ROM to obtain a value for the desired pan angle. To find the tilt angle, it is first necessary to establish the radial position of the target point in the transformed horizontal plane by calculating the square root of the sum of the squares of the co-ordinates x' and y'. In carrying out this calculation it is necessary to detect an overflow condition which exists if the sum of the squares is a 33 bit number. If this condition is detected, each square is divided by four and a new sum is formed, an overflow flag being set to indicate that overflow has occurred. The square root is found by up to sixteen steps of successive approximation and the result is doubled if the overflow flag was set during the calculation. The resulting square root is divided by the value z' and the result is applied as before to the ARCTAN table to determine the tilt angle. The results obtained represent the new pan and tilt positions to which the lamp is to be moved.

The arrangement described for sending out x, y and z co-ordinate data instead of pan and tilt angle data is highly advantageous in that it enables the console main cpu load to be significantly reduced and also makes it very easy for a console operator to control light beam movements. It is frequently required for a group of lamps to be used together to illuminate a single performer. Where the performer moves from one position on stage to another it is required for all the lamps to change position simultaneously to follow. If the system involved transmission of pan and tilt angle data, this data would be different for every lamp in the group. It would have to be set up by the console operator and stored in cue files on the hard disk drive unit **15**. This would be a very time consuming operation as the pan and tilt angles for each lamp would have to be established and recorded individually. The cue record would need to be of considerable size to record all the different data for each lamp. With the arrangement described above, however, only the x,y,z co-ordinate data needs to be stored and when the cue is recalled the same data is sent to each of the lamps in the group.

Whilst it is theoretically possible to use stored cue data in x,y,z coordinate form and to use the console main cpu **14** to calculate the pan and tilt angles to send to the lamps, this would be unsatisfactory as the calculations involved would impose a very heavy load on the cpu **14**, particularly where a large number of lamps in several different groups had to be moved as the result of a single cue.

As described above a "point-at" mode is envisaged as the normal operating mode. However, other modes of operation are also envisaged. For example, the lamp could be instructed to point away from the point specified or to point in a direction parallel to a line joining a fixed point (eg the origin of the co-ordinate system) to the point specified. These "point-away" and "point parallel" modes would be selected by means of flags included in the data transmitted to the lamps.

The arrangement described enables the lamps to be very precisely synchronised. The data is transmitted from the distribution unit to all of the lamps simultaneously and each lamp can start to respond at the end of the message. This enables very precise direction of all the lamps to a moving point in "point-at" mode and very clean parallel sweeps to be made in "point parallel" mode.

It should be noted that the use of x,y,z co-ordinates is also very advantageous in situations where a pre-arranged lighting performance is to be used in several different venues. The pre-loaded gantries or trusses used for such touring performances cannot always be mounted at exactly the required positions relative to the stage because of local conditions. In this case all that is needed is for offsets data to be sent to the lamps at set-up time to enable each lamp cpu to correct its position data. No editing of the individual pre-recorded cues is necessary as it would be in the same circumstances if pan and tilt data were stored.

As part of the set-up procedure for each performance it is necessary to initialise the values of the actual pan and tilt angle count-states, since encoders of the type used do not give any absolute position data. This is accomplished by driving the lamp to an end stop in one direction for each movement. The lamp is driven back to a predetermined number of counts and the counters are reset to zero at this position.

Turning now to FIGS. **5** to **7**, the circuitry for controlling the individual dc servo-motors inside the lamp is more complex as each co-processor has to deal with up to six servo-motors. As shown in FIG. **5**, the co-processor **28** controls a number of data routers **50** to **54** which determine which channel is being controlled at any given time. The router **50** co-operates with six HCTL-2016 counters **55** which count the quadrature pulse outputs of the respective encoders, to determine which of the counters should supply its count-state to the co-processor **28**. Router **51** controls individual resetting of the counters **55**. Router **52** co-operates with a 74HC175 ic **56** (one for each channel) to determine which **L6202** ic motor controller **57** is enabled and also routes "RIGHT" and "LEFT" signals from the co-processor to the circuits **57**. Router **53** controls routing of position error data calculated by the co-processor **28** for each channel to latches **58** (one for each channel) at the input of pulse width modulator circuits for controlling the motor controllers **57**. This error data is actually passed to the latch **58** in an inverted form, so that the larger the error, the smaller the value passed is. Router **54** routes various digital sensor signals to a sensor input of the co-processor, Such sensors are utilized by some of the channels to indicate when the moving part in question is in a datum position. This is required for the gobo wheels, the colour wheels and the shutter, but not for the iris diaphragms or lenses which can be moved to end stop positions. During datum set-up the sensors (optical sensors sensing a hole or flag or Hall effect sensors) are detected and the HCTL counters are reset.

As co-processor **28** has only 256 bytes of internal memory, extra memory is required for each channel to store program variables. The RAM selection control circuit is shown in FIG. **7**. The memory ic **59** (an HM6116LP ic) has **11** address lines of which eight are connected to the co-processor write bus via a latch circuit **60** and the remaining three of which are connected to spare outputs of three of the ics **56**. Spare outputs of the selectors **50, 51, 52** are connected to control terminals of the memory ic and a spare output of the selector **53** is connected to an output enable terminal of the latch circuit **59**. Thus a particular address in the memory ic can be selected by the co-processor by first setting the ics **56** and the selectors **50, 51, 52** to appropriate states and then outputting the lower bytes of the address to latch **60** whilst output from latch **60** is enabled. Two further eight bit latches **61** and **62** provide temporary storage for data to be written to and data just read from the memory ic **59**. When neither reads nor writes are required the memory data bus is tri-stated. Bus contention is thus avoided.

Circuit **57** actually controls the motor current, but it in turn is controlled by a pulse width modulator circuit, comprising the latch **58** and a digital comparator **65** which compares the contents of latch **58** with the count-state of an 8-bit continuously running counter ***66a*, *66b*** serving all channels. The comparator output goes high when the count-state exceeds the latch contents, so that if the latch content is low the comparator output is high for a high proportion of each cycle of the counter ***66a*, *66b***. The output of the comparator **65** is ANDed with an enable output from ic **56** by a gate **67** and then with the output of an overcurrent detector circuit **68** by another gate **69**.

When a new target value for one of the parameters controlled by co-processor **58** arrives in the receive buffer, and it is associated with execution duration data (this may apply to lens movements, colour changer movements, gobo movements and iris diaphragm movements, but not shutter movements) the cpu **21** handles time slicing as in the pan and tilt operations. Since several channels are controlled by each co-processor, however, no interpolation by the co-processor is used. Instead each channel has its error checked and a new value written (if necessary) to latch **58** every 12 mS.

In the case of the shutter, the message received by the lamp merely includes a shutter open or shutter closed command. When the required shutter status changes, the main cpu merely increases the target shutter angle by 45 degrees (in the case of a four bladed shutter) and passes the new value to the co-processor.

This arrangement enables the shutters of some or all of the lamps to be operated in synchronism. Moreover, the console cpu **14**, can operate to update the shutter open/closed instructions at regular intervals to obtain a stroboscopic effect, synchronised for all the lights.

We claim:

1. A communication system for communicating between a controlling unit and at least one lamp unit controlled by the controlling unit over a serial link, the communication system comprising:

a) a central controller unit, having:

a line driver, connected to the serial link, for producing output signals when enabled and driven by an input signal;

a controlling processor, monitoring status indicative of messages to be sent to at least one lamp unit, producing said input signal for said line driver indicative of said messages, and controlling said line driver to selectively enable and disable said line driver, said processor including an associated timer, which causes a time delay of a first predetermined time between enabling said line driver and producing said input signal; and

b) a lamp unit having:

a line receiver, connected to said serial link to receive serial communication information therefrom;

a buffer for holding information from said line receiver; and

a processor which inspects contents of said buffer, said processor including a timer which produces interrupts which are synchronized with initial receipt of information and at a second predetermined time thereafter, the second predetermined time being shorter than the first predetermined time so that any initial spurious information sent on enable of said line driver always causes a time-out of said processor in said lamp.

2. A communication system as in claim **1** wherein said serial link links one port of said central controller unit with only a single lamp unit.

3. A communication system as in claim **1** wherein said first predetermined time is a time necessary for receiving two bytes, and said second predetermined time is between the respective times necessary to transmit one byte and two bytes.

4. A method of communicating between a controlling unit and at least one lamp unit controlled by the controlling unit over a serial link, the method comprising the steps of:

coupling a line driver to the serial link;

controlling said line driver to produce output signals when enabled and driven by an input signal;

monitoring status indicative of messages to be sent to at least one lamp unit, and producing said input signal for said line driver indicative of said messages,

controlling said line driver to selectively enable and disable said line driver,

causing a time delay of a first predetermined time between enabling said line driver and producing said input signal;

connecting a line receiver to said serial link to receive serial communication information therefrom;

producing interrupts which are synchronized with initial receipt of information at said line receiver and at a second predetermined time thereafter, the second predetermined time being shorter than the first predetermined time so that any initial spurious information sent on enable of said line driver always causes a time-out of said processor in said lamp.

5. A method of moving a controlled object to arrive at a desired point at a desired time, comprising the steps of:

determining a number of time-sliced intervals, each of a constant time, which will occur between a start time, and a complete time at which the movement is to be completed;

defining a travel parameter, and incrementing said travel parameter by an amount equal to a reciprocal of the number at each time-slice interval to produce an incremented travel parameter;

multiplying said incremented travel parameter by a total distance to be travelled and using that value to produce a desired current distance value for the present time-slice; and

continuing said incrementing and multiplying until the travel parameter reaches unity or above.

6. A method as in claim 5 comprising the further step of imposing a sinusoidal profile on the travel parameter.

7. A method as in claim 5, wherein said controlled object is an x, y, z-controlled lamp.

8. A method of moving a controlled object to arrive at the desired point at a desired time, comprising:

determining intervals at which time slicing should occur, said interval being a slice of time, and each occurring between a start time and a completion time at which movement is to be completed;

defining a parameter of movement of said controlled object, said parameter of movement defining a position of said controlled object;

incrementing said parameter at each slice of time by an amount inversely proportional to said time slice interval and directly proportional to a total distance to be travelled to form a new travel parameter;

using the new travel parameter to produce a desired current value for position at a present time; and

continuing said incrementing and using until the controlled object arrives at the desired point.

9. A method as in claim 8 further comprising imposing a sinusoidal profile on the travel parameter.

* * * * *

EXHIBIT C

US005769531A

United States Patent [19]

Hunt et al.

[11] **Patent Number: 5,769,531**

[45] **Date of Patent: Jun. 23, 1998**

[54] **STAGE LIGHTING LAMP UNIT AND STAGE LIGHTING SYSTEM INCLUDING SUCH UNIT**

[75] Inventors: **Mark Alistair Hunt**, Derby; **Keith James Owen**, Moseley; **Michael Derek Hughes**, Wolverhampton, all of United Kingdom

[73] Assignee: **Light & Sound Design, Ltd.**, Birmingham, England

[21] Appl. No.: **448,861**

[22] Filed: **May 24, 1995**

Related U.S. Application Data

[62] Division of Ser. No. 77,877, Jun. 18, 1993, Pat. No. 5,502,627.

[30] **Foreign Application Priority Data**

Sep. 25, 1992	[GB]	United Kingdom	9220303
Sep. 25, 1992	[GB]	United Kingdom	9220309
Apr. 20, 1993	[GB]	United Kingdom	9308071

[51] **Int. Cl.**[6] **F21M 1/00**

[52] **U.S. Cl.** **362/233**; 362/286; 362/386

[58] **Field of Search** 362/233, 285, 362/286, 269, 271, 272, 385, 386

[56] **References Cited**

U.S. PATENT DOCUMENTS

4,837,665	6/1989	Hoyer et al.	362/233 X
4,980,806	12/1990	Taylor et al.	362/233 X
5,045,983	9/1991	Shields	362/293

FOREIGN PATENT DOCUMENTS

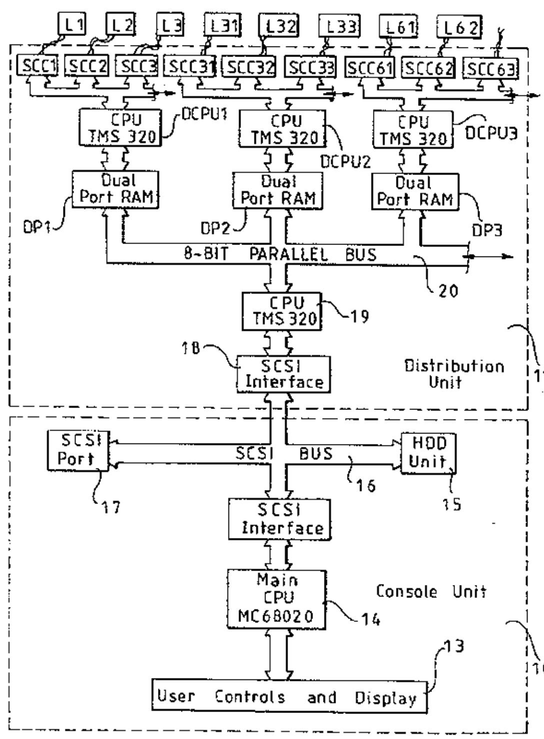
0495305	7/1992	European Pat. Off. .
2643134	8/1990	France .
2209229	5/1989	United Kingdom .
2231138	11/1990	United Kingdom .
8905421	6/1989	WIPO .

Primary Examiner—Stephen F. Husar
Attorney, Agent, or Firm—Fish & Richardson P.C.

[57] **ABSTRACT**

A stage lighting lamp unit includes a processor for receiving control data from a remote console. Beam orientation data for the lamp unit is passed to the lamp in the form of the x, y and z co-ordinates of a point in space through which the beam is to pass. The processor divides the required lamp travel into a number of stages dependent on execution duration data sent with the position data, and calculates, for each stage, a new value for pan and tilt angles for the lamp. These values are passed to pan and tilt controlling co-processors which control servo-motors for pan and tilt operation. The lamp unit also incorporates a rotatable shutter for interrupting the lamp beam when required. The shutters of all the lamps in a system can be instructed from the remote console to open and close in synchronism, thereby providing a stroboscopic effect.

13 Claims, 10 Drawing Sheets



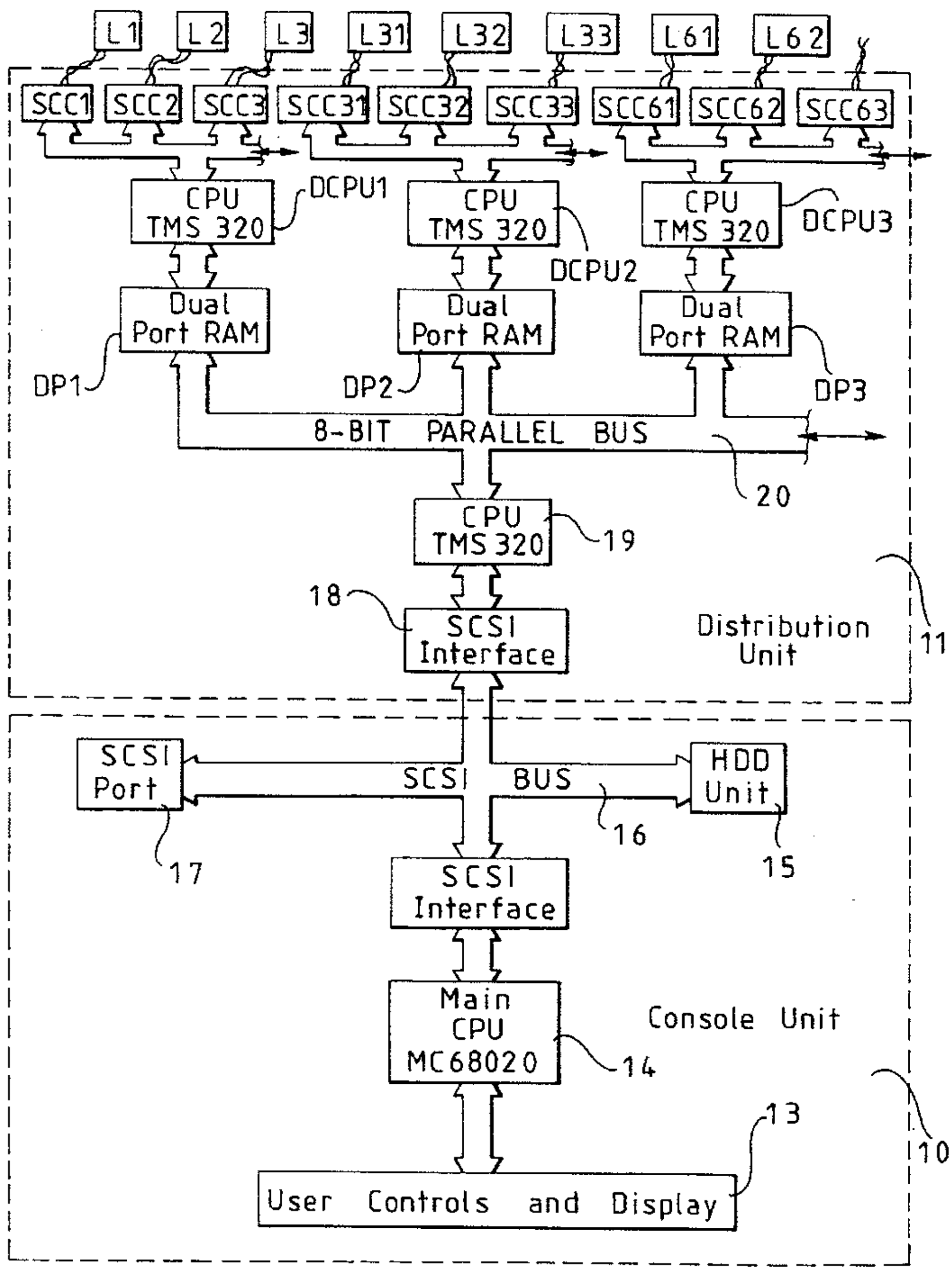
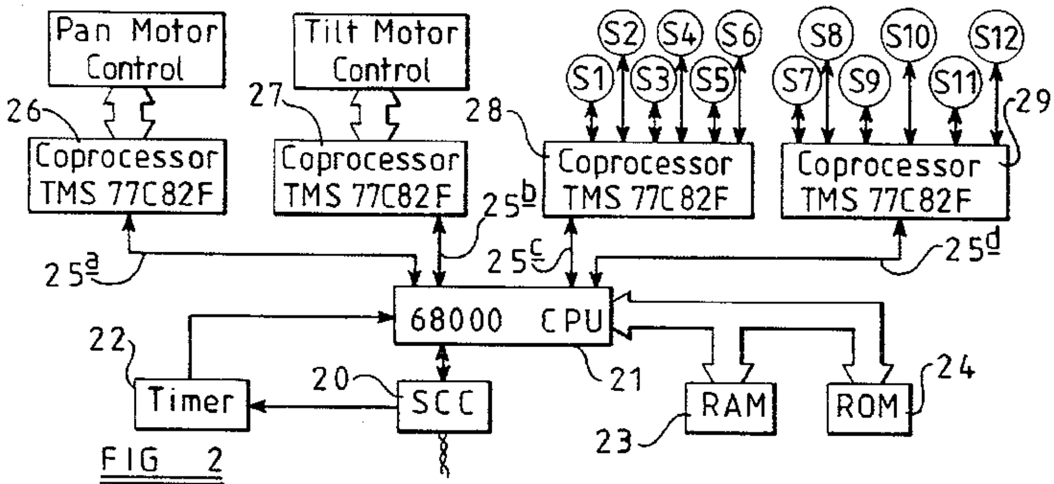
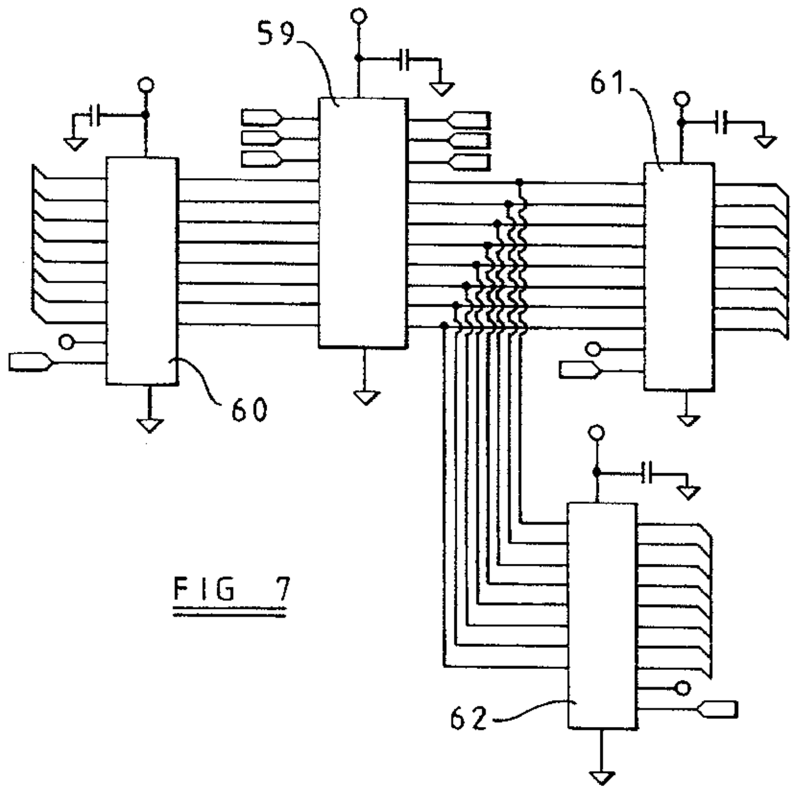

L1
L2
L3
L31
L32
L33
L61
L62
SCC1
SCC2
SCC3
SCC31
SCC32
SCC33
SCC61
SCC62
SCC63
CPU TMS 320
DCPU1
CPU TMS 320
DCPU2
CPU TMS 320
DCPU3
Dual Port RAM
DP1
Dual Port RAM
DP2
Dual Port RAM
DP3
8-BIT PARALLEL BUS
20
CPU TMS 320
19
18
SCSI Interface
Distribution Unit
11
SCSI Port
SCSI BUS
HDD Unit
16
17
15
SCSI Interface
Main CPU MC68020
Console Unit
14
13
10
User Controls and Display

FIG 1



Pan Motor Control
Tilt Motor Control
Coprocessor TMS 77C82F
Coprocessor TMS 77C82F
Coprocessor TMS 77C82F
Coprocessor TMS 77C82F
S1
S2
S3
S4
S5
S6
S7
S8
S9
S10
S11
S12
68000 CPU
Timer
SCC
RAM
ROM
26
27
28
29
25a
25b
25c
25d
20
21
22
23
24

FIG 2



59
60
61
62

FIG 7

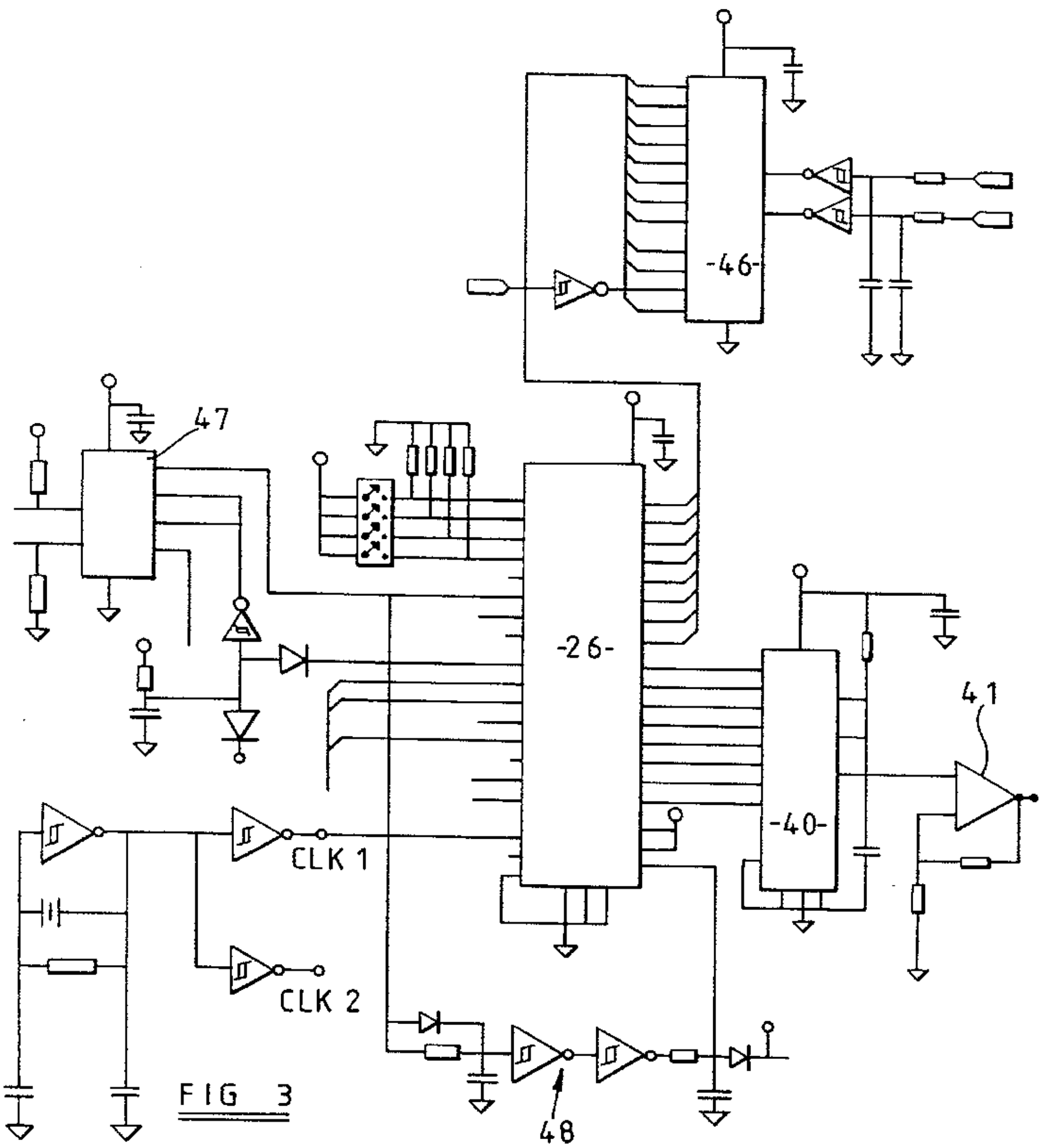


FIG 3

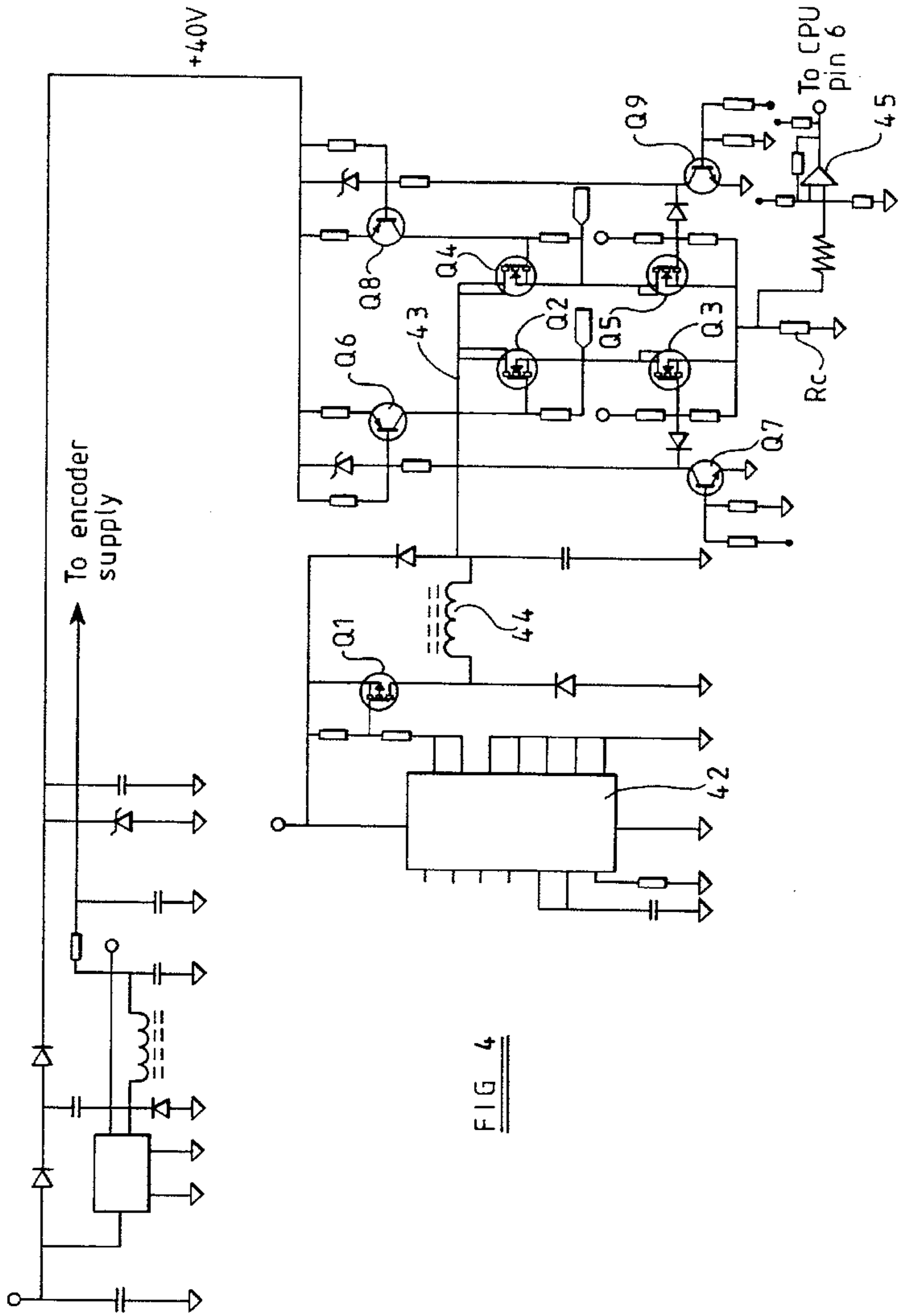

+40V
To CPU pin 6
Q9
45
Q8
Q4
43
Q6
Q2
Q5
Q3
Rc
Q7
To encoder supply
Q1
44
42

FIG 4

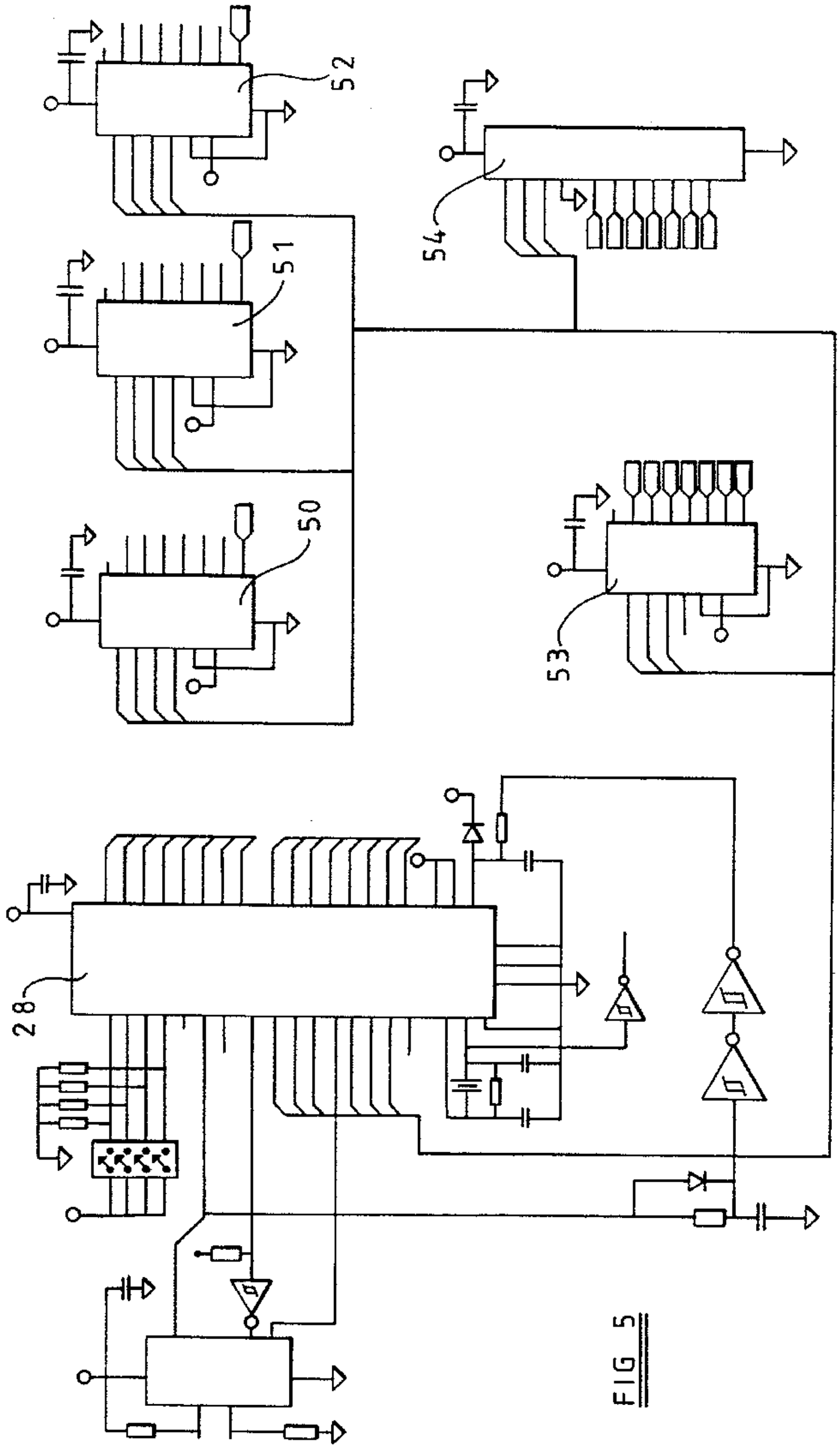


FIG 5

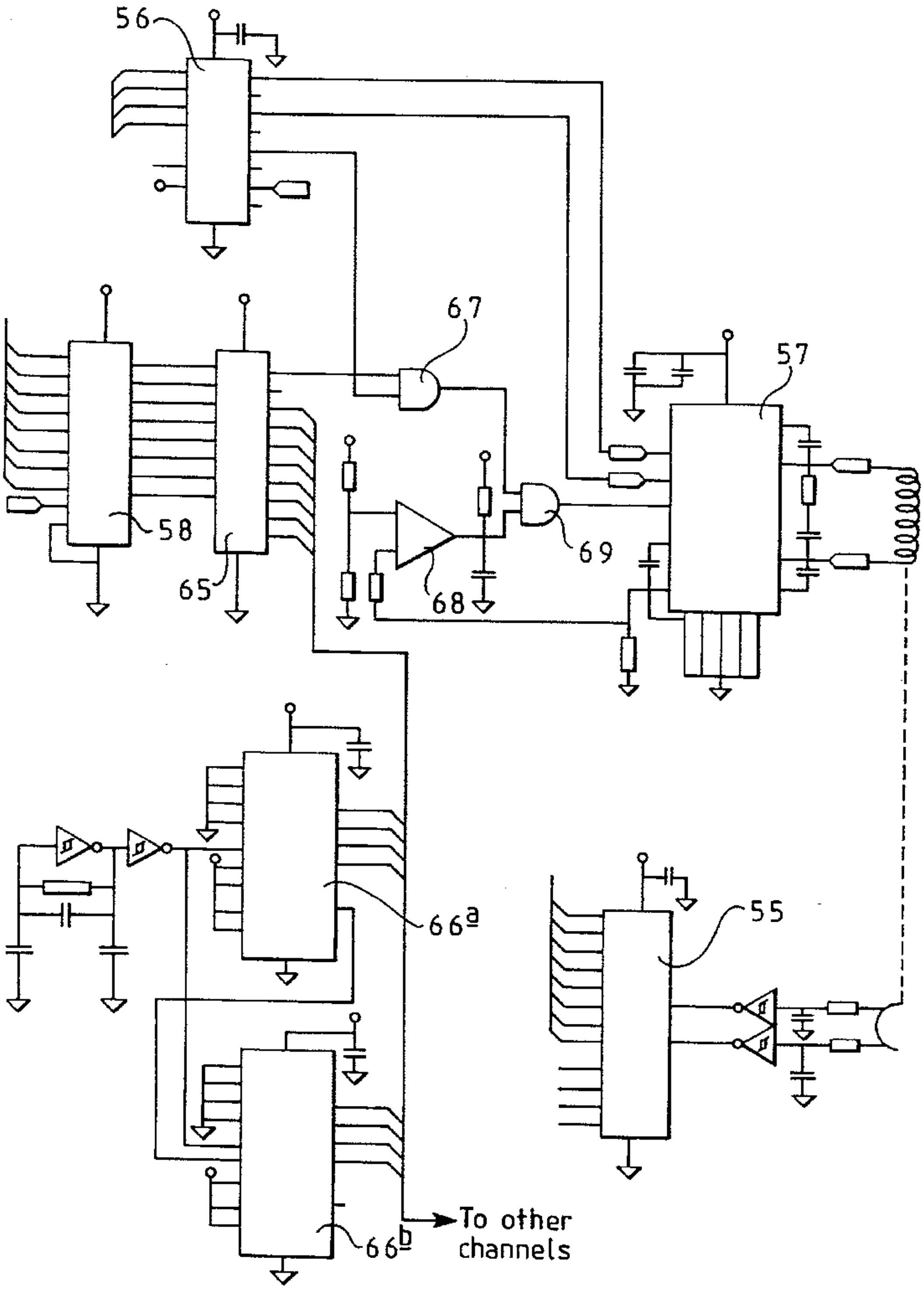


FIG 6

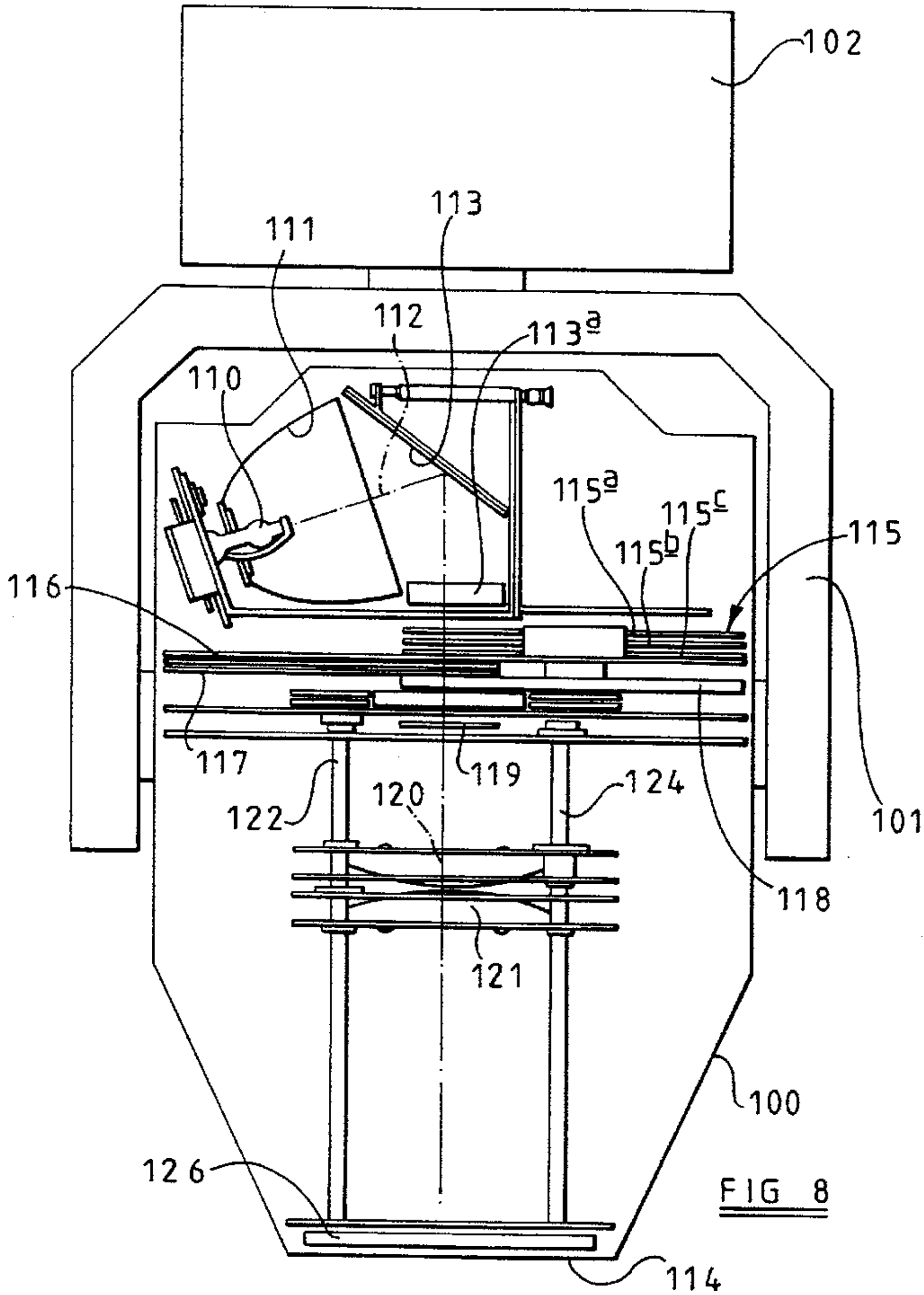

102
113
111
112
113a
110
115a
115c
115
115b
116
117
119
124
122
120
101
118
121
100
126
114

FIG 8

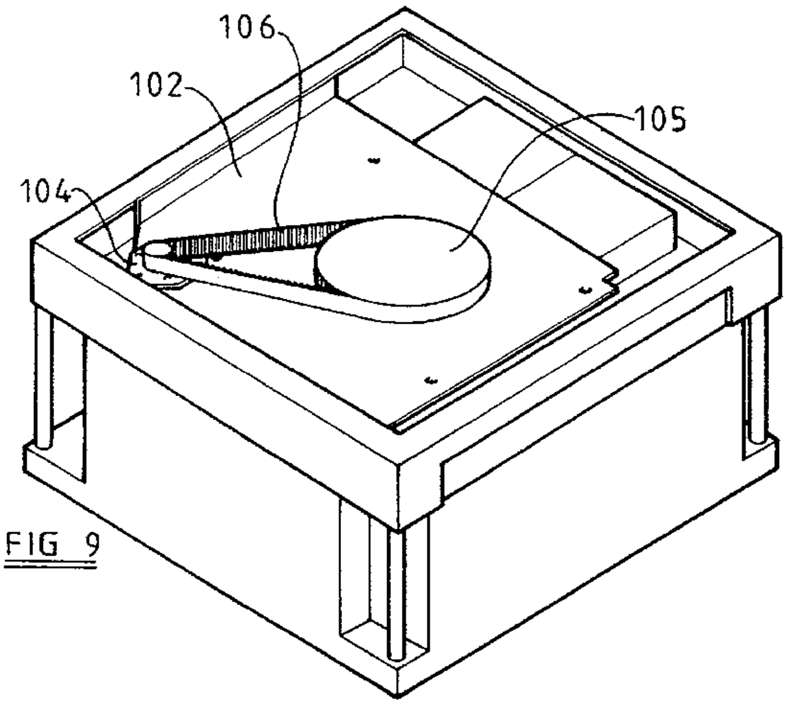


FIG 9

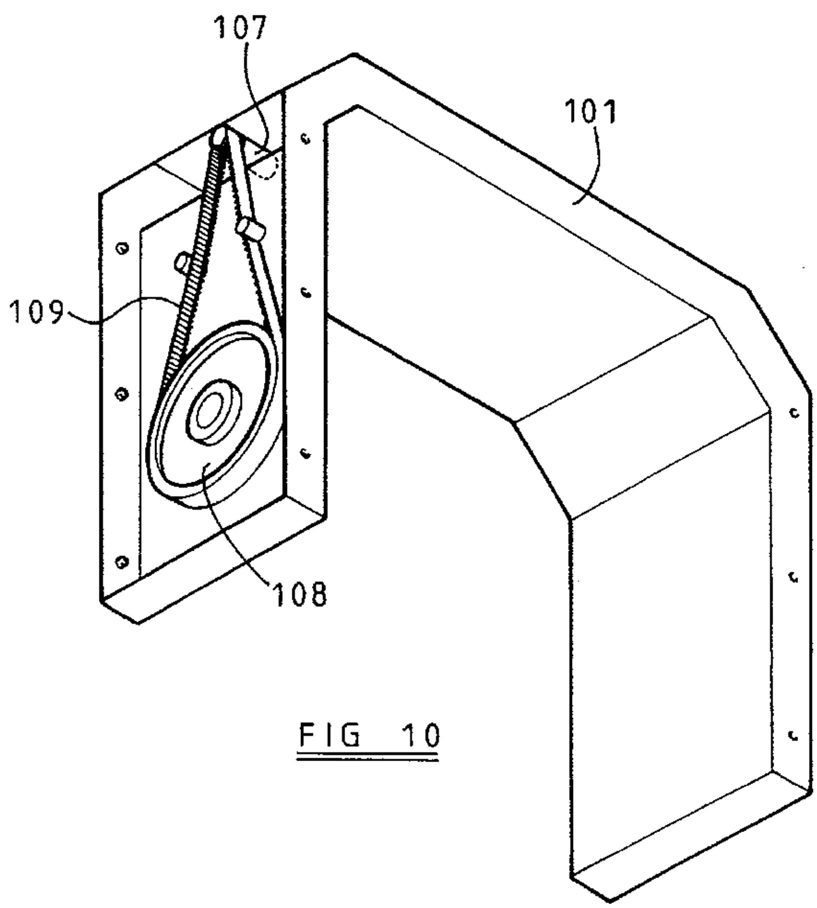


FIG 10

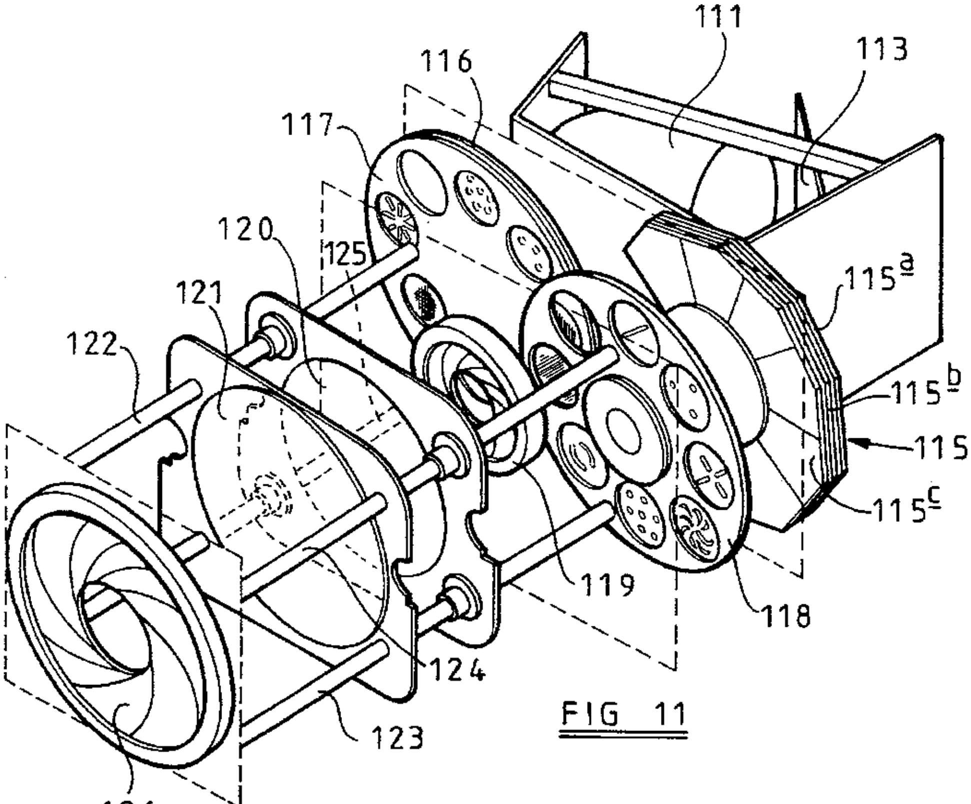

111
113
116
117
120
125
121
122
115a
115b
115
115c
119
118
124
123
126

FIG 11

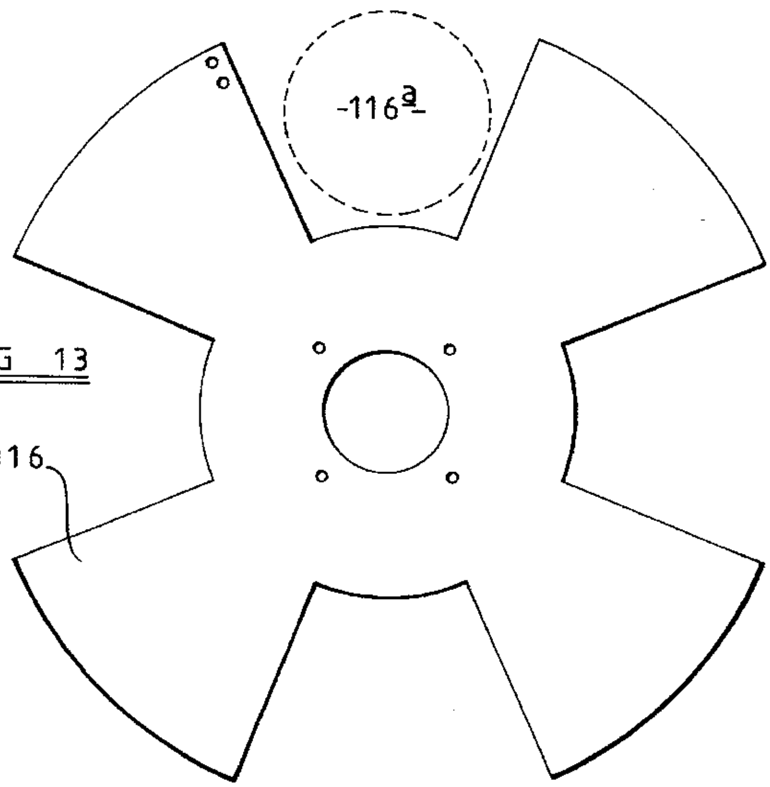

116a
116

FIG 13

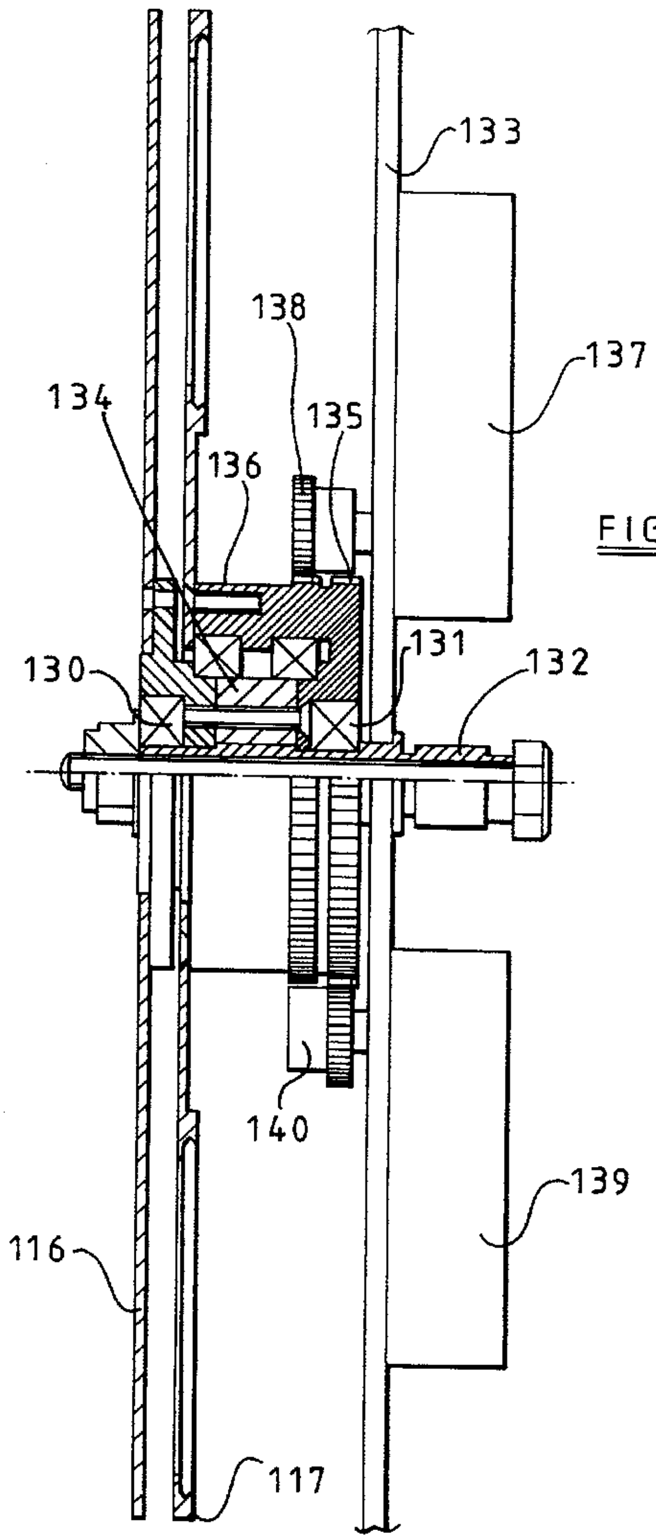


FIG 12

STAGE LIGHTING LAMP UNIT AND STAGE LIGHTING SYSTEM INCLUDING SUCH UNIT

This is a divisional of application Ser. No. 08/077,877, filed Jun. 18, 1993, now U.S. Pat. No. 5,502,627.

This invention relates to stage lighting and is particularly concerned with the control of multiple functions of a lamp.

It has already been proposed to incorporate in a lamp unit a plurality of different functions, such as colour changers, focusing lenses, iris diaphragms, gobo selectors and pan and tilt mechanisms which are controlled from a remote console. Stage lighting systems have as a result reached very high levels of complexity requiring a very complicated main control console and lamp unit constructions. The use of microprocessors, both in the console and the lamps has become conventional as increasing complexity makes it more difficult to produce and subsequently maintain a system which uses hard wired logic or analog controls. In such systems the microprocessor in the console is used to allow the user to set up lighting cues and to control the sending of appropriate data to the lamp microprocessors. The lamp microprocessors are also involved in controlling communication between the console and the lamps, and also have to control a plurality of servo-motors which drive the various functions of the lamps.

It is one object of the present invention to provide a lamp micro-processor and servo-control arrangement which allows complex functions to be carried out.

It is another object of the invention to provide a lamp control system in which control of pan and tilt movements of each lamp can be carried out in rapid and efficient manner, enabling large groups of lamps to make co-ordinated movements.

It is yet another object of the invention to provide each lamp in a stage lighting system with a means for quickly interrupting its light beam and quickly re-establishing the beam so that a group of lamps can be made, when required to flash in synchronism.

In accordance with one aspect of the invention there is provided a lamp unit for connection to a remote control console for the control of a plurality of different functions of the lamp, said unit comprising a main processor circuit, associated with a communication controller for accepting message data from the console, a plurality of servo-controls for operating said functions of the lamp, and a plurality of co-processors which are connected to the main processor circuit so as to be supplied thereby with desired value data for the various lamp functions, said servo-controls being controlled by said co-processors.

In the case of pan and tilt controls where close control is required throughout the movement of the lamp from an initial position to a new position, one of the co-processors is assigned solely to the control of movement about each axis. Other functions can share a co-processor.

The main processor circuit of the lamp is preferably programmed to accept data from the control console defining not only a target position for any function, but also a duration over which the function is to be executed. In this case the main processor circuit divides the "journey" into segments and updates the target position data passed to the associated co-processor at intervals.

In accordance with another aspect of the invention, there is provided a lighting control apparatus comprising the combination of a main control console for accepting user input relating to required beam movements, a plurality of independently operable lamp units situated remotely from

the console, each of the lamp units incorporating a servo-mechanism for automatically moving the lamp beam about two mutually transverse axes to a desired angular position and data communication means connecting the console to the lamp units for the transmission of desired position data to the lamp units, the desired position data being transmitted in the form of a set of three dimensional linear co-ordinates defining a point in space through which the lamp beam is required to pass, and each lamp unit including a calculating device for calculating the desired angular position from the desired position data and supplying the servo-mechanism with such desired angular position.

In addition to the "point at" mode of operation mentioned above, additional modes may be specified in which the lamps point away from the specified point or in which they all point in the same direction parallel to a line between a fixed position in the co-ordinate system and the specified point.

Conveniently, all the data concerning the positions and orientations of the individual lamp units within the co-ordinate system is stored in a setup file kept on a hard disk drive in the console. When the same lighting set-up is used at different venues, where it is impossible to set the frame which carries all the lamp units at exactly the same position as that for which the set-up was designed, offset data can be input at the console and either used within the console microcomputer to correct the position data stored during set-up as it is sent out, or such data can be sent to all of the lamp units over the network and stored there, to enable the corrections to be made in the individual lamp processor units.

In accordance with another aspect of the invention, a stage lighting unit comprises a housing, a light source within said housing, an optical system for forming light from said light source into a beam, a rotary shutter device having a plurality of blades, said shutter device being rotatably mounted in the housing so as to cause said blades to pass through and obstruct said beam as the shutter device rotates, a motor for rotating said shutter device and a servo-control for controlling said motor in accordance with data received in use from a remote control console.

The invention also resides in a stage lighting system incorporating a plurality of lighting units as defined above controlled by a common remote control console via data communication means, whereby the rotary shutter devices of all the units can operate in synchronism.

An example of the invention will now be described with reference to the accompanying drawings, in which:

FIG. 1 is a block diagram of a stage lighting system;

FIG. 2 is a block diagram of the internal circuitry of one of a plurality of lamp units in the system of FIG. 1;

FIGS. 3 and 4 are more detailed circuit diagrams showing a pan motor drive control forming part of the internal circuitry of the lamp;

FIGS. 4 to 7 are detailed circuit diagrams showing a rotary shutter motor drive control forming part of the internal circuitry of the lamp;

FIG. 8 is a diagrammatic, part-sectional view of one of the lamps;

FIG. 9 is a perspective view of a pan movement drive arrangement;

FIG. 10 is a perspective view of a tilt movement drive arrangement;

FIG. 11 is a diagrammatic perspective view of the internal moving parts of the lamp;

FIG. 12 is a sectional view showing the drive arrangement for a shutter and a gobo wheel forming part of the lamp; and

FIG. **13** is an elevation of a shutter wheel forming part of the lamp.

Referring firstly to FIG. **1**, the system consists basically of a console unit **10**, a signal distribution unit **11** and a plurality of lamps L1, L2, L3 . . . , L31, L32, L33 . . . , L61, L62 . . . individually connected by twisted pair data communication links to the distribution unit.

The console unit **10** has an array of switches, slider potentiometers, rotary digital encoders and other user actuable input devices (not shown) and a display indicated at **13**. These are all connected to main console cpu **14** (an MC68020 micro-processor) which has the task of receiving inputs from the user actuable input devices and controlling the display. Both tasks are assisted by separate co-processors which directly interface with different parts of the console.

The main cpu can communicate with a hard disk drive unit **15** via a SCSI bus **16** which also connects it to the distribution unit and to an external SCSI port **17**, through the intermediary of which the console can, if required be connected to a personal computer. The user controls can be used in setting up a sequence of cues in advance of a performance, the sequence being stored in a cue file on the hard disk drive unit **15**. The sequence can be recalled during the performance to enable the various stored cues to be executed. Direct manual control of the lamps from the console is also possible as is manual editing of cues called up from the hard disk. The main console cpu **14** creates messages to be sent to the individual lamps, each message comprising a fixed number of bytes for each lamp. The messages contain data relating to the required lamp orientation, beam coloration, iris diaphragm diameter, gobo selection and rotation, zoom projection lens control and opening or closing of a shutter included in the lamp. A block of the RAM of the main cpu is set aside for the storage of these messages, the block being large enough to contain messages for 240 lamps, being the largest number which can be controlled via the distribution unit. Where it is required to control more than 240 lamps additional distribution units can be connected to the SCSI bus and extra main cpu RAM reserved for message storage. When any message data is changed the main cpu **14** sets a flag in the RAM block which is detected at a given point in the main cpu program loop and interpreted as a signal that the changed message data is to be transferred to the distribution unit **11**.

The distribution unit **11** has a main cpu **19** which controls reception of data from the SCSI bus interface and distribution of such data to up to eight blocks of dual,port memory DP1, DP2, DP3 . . . via an eight bit data bus **20**. The cpu **19** is alerted to the waiting message data when cpu **14** selects the distribution unit. The cpu **19** then supervises byte by byte transfer of the message data which it routes to the various blocks of dual port memory.

For actually sending out the message data to the lamps, there are a plurality of serial communication controllers SCC1 to SCC**30**, SCC**31** to SCC**60** etc, there being thirty serial communication controllers associated with each block of dual port memory. A further cpu DCPU1, DCPU2, etc is associated with each block of dual port memory and distributes message data transferred to the dual port memory to the individual serial communication controllers and the messages are transferred to the lamps. Each serial communication controller in the distribution unit includes a line driver which can be disabled except when data is to be transmitted. Enabling of the driver can cause a spurious signal to be transmitted over the data link. To allow such spurious signals to be identified and ignored, a two-byte gap is left between enabling the line driver and commencing transmission of the message data for the channel in question.

This will be described in more detail herein. All asynchronous serial communication systems require framing information to synchronize the reception process. This has been typically done in the prior art using start bits and stop bits.

The present invention preferably uses FM**0** coding in which the data is transmitted as one cycle of the carrier frequency for a zero or as a half cycle of the carrier frequency for a one. When the line has been idle, no waveform at all is present. When the line drivers are first enabled, an arbitrarily short pulse will usually appear on the line, due to lack of synchronization between the data signal and the enabling signal. This short data pulse could be misinterpreted as a start bit, for example and if so it would disturb later framing.

The present invention avoids any problems from this arbitrarily short pulse. To avoid this, the present invention uses a timer on the receive line, set to the time needed to receive two bytes on the serial data line. This timer is restarted whenever a byte on the data line is detected.

Each time the timer interrupt occurs, the number of bytes received is checked against the number of bytes in a valid data frame. If the number is incorrect, then the count is cleared and the message is discarded. If correct, the information is passed to the main program loop by setting a flag variable.

When the data line is first enabled, the distribution box has an internal delay of at least two byte times, which must elapse before any data will be sent. Any data received by the lamp will therefore be discarded as noise by the timer interrupt routine. After that, the real data can be safely sent down the line since the start bit of the first byte will be received correctly. When the transmission is completed, the line drivers will be disabled again.

Each of the cpus eg DCPU1, transfers data from the associated dual port RAM DP1 to the serial communication controller SCC1 to SCC**30** with which it is associated one byte at a time, ie the first byte for SCC**1** is transferred followed by the first byte for SCC**2** and so on, each serial communication controller commencing transmission as soon as it has received its byte of data. The serial communication controllers operate to transmit data at 230.4 Kbps so that it takes about 35 μs to transmit each byte. Transfer of data from the dual port RAM DP1 to the serial communication controllers is, however, at a rate of several Mbps, so that the transmissions from all the serial communication controllers are almost simultaneous. The cpu DCPU1 is not required to monitor the transmission of data by the serial communication controller, but utilizes a software timer to commence transfer of the second byte to the serial communication controllers. This timer is started when transfer of the byte of data to the last serial communicationcontroller SCC**30** has been completed and its time-out duration is slightly longer than the byte transmission time, say 40 μs. Transmission of all the messages takes about 1.5 ms out of a distribution unit main program loop duration of 4 ms.

As shown in FIG. **2**, each lamp includes a serial communication controller **20** which controls reception of message data from the individual data link connecting it to the distribution unit **11**. The receipt of any signal from the data link causes an interrupt of the lamp main cpu **21** (another MC68000) and the cpu **21** then controls acceptance of the signals. A timer **22** times the gaps between bytes received from the data link and this timer causes another interrupt on time-out. The time-out time of the timer is between the times taken to transmit 1 and 2 bytes, so that time out always occurs following a spurious signal caused by line driver

enabling. The timeout interrupt causes the cpu **21** to inspect the total number of bytes received since the initial interrupt and if this is less than the expected number of bytes (which is constant) the message is ignored. The time-out interrupt also resets a software data pointer to the beginning of a receive buffer in readiness for the next transmission.

The cpu **21** operates in accordance with programs stored in the lamp cpu ROM. On receipt of a message of valid length, a program variable representing the number of messages received since the lamp program was last started is incremented and the main program loop of the lamp cpu checks this variable every 16 mS. If the variable has changed since the last check, the data in the receive buffer is compared with corresponding values of variables representing current "desired values" of the various lamp function parameters. For example the receive buffer may contain two bytes representing the x, y and z co-ordinates of a point in an orthogonal three dimensional frame of reference, through which point it is required that the axis of the lamp beam should be directed. If the values of the corresponding byte pairs in the receive buffer and the desired value variables already contained in the cpu RAM are the same, no action is taken in respect of the control of the motors which control pan and tilt action of the lamp (to be described in more detail hereinafter).

As shown in FIG. **2**, the main lamp cpu **21** communicates via serial data links **25***a*, **25***b*, **25***c* and **25***d* with four servo-control co-processors **26**, **27**, **28** and **29**. Each of these co-processors is a TMS77C82 cpu. Co-processors **26** and **27** respectively control pan and tilt operation, and each of the co-processors **28** and **29** can control up to six different dc servo-motors operating different functions of the lamp.

Before proceeding with a more detailed description of the circuitry and operation of the lamp electronics, some detail will be given of the various functions of the lamp. FIG. **8** shows the relative positions of a plurality of independently operable beam characteristic control elements within the lamp housing **100**. The lamp housing is pivotally mounted on a U-bracket **101**, which is itself pivotally mounted on a mounting base **102**. FIG. **9** shows the mounting base **102** which incorporates a pan drive motor/gearbox/optical encoder arrangement **104** which drives a gear **105** attached to the U-bracket via a reduction toothed belt drive **106**. FIG. **10** shows how, within the hollow structure of the U-bracket **101**, there is mounted a tilt drive motor/gearbox/optical encoder **107** which drives a gear **108** attached to the lamp housing via another reduction toothed belt drive **109**.

As shown in FIGS. **8** and **11**, within the lamp housing, a light source **110** is mounted within an ellipsoidal reflector **111** providing a light beam with an axis **112** which is reflected by a mirror **113**, which is a dichroic mirror that reflects only visible light and passes ultra voilet and infra red light, the reflected light passing out through an opening **114** at the opposite end of the housing. The reflector **111** has a generally cup-shape surrounding the bulb **110**. According to one aspect of the invention, the axis **112** has an angle pointing in a direction rearward relative to a perpendicular to the central axis **120** of the lamp unit. If the reflector is located as shown, such that an outside edge of the reflector is generally parallel to a rear end of the housing, the optimal packing efficiency is achieved. As shown in FIG. **8**, this allows the reflector to be most efficiently packed into the available space. The reflected beam from the mirror **113** passes firstly through a collimating lens **113***a*, and then the colour changer **115** which comprises dichroic filters having differing transmission characteristics mounted on co-centered three filter disks **115***a*, **115***b* and **115***c* rotable

around a common axis of rotation. Each disk has nine different filters on it and one blank space around its periphery, so that up to 1000 different combinations of filters can be positioned across the beam by selective positioning of the three disks (although not all of these combinations are necessarily useful as some may block all visible light). The blank space of each of the disks can be used to eliminate any color changing characteristic of that disk. These disks are driven by three of the dc servo-motors. Next the light beam passes through the plane of a bladed shutter **116** (shown in FIG. **13**) and a first gobo wheel **117** which has various gobos mounted in or over circular holes therein. As shown in FIG. **12** described in more detail hereinafter, two motors are committed to driving the shutter **116** and the gobo wheel **117** respectively. Next, there is a second gobo wheel **118** on which there are mounted a plurality of gobos which are rotatable relative to the wheel **118**. There is one motor (not shown) for driving the gobo wheel **118** and another for rotating the gobos mounted thereon through a gear arrangement (not shown). Next along the light beam is a beam size controlling iris diaphragm **119** driven by another motor (not shown). Two further motors (not shown) drive two lens elements **120**, **121** along guides **122**, **123** parallel to the beam axis using lead screws **124**, **125**. The lens elements form a simple two element zoom lens controlling the spread and focus of the beam. Finally, an outer iris diaphragm **126** is provided adjacent the opening **114** and this is driven by a further motor (not shown). In the example described, therefore only eleven channels are actually employed.

Referring now to FIG. **12**, the shutter **116** is rotatably mounted on bearings **130**, **131** on a shaft **132** fixed to a mounting panel **133** which is secured to the housing. The gobo wheel **117** is rotatably mounted on bearings on a tubular shaft **134** which acts to space the shutter **116** from a first drive gear **135**. The gobo wheel **117** is actually mounted on a second drive gear **136**. The shutter motor **137** (which is combined with a reduction gearbox and an optical encoder) is mounted on the panel **133** and drives a pinion **138** meshed with the first gear **136**. Similarly motor **139** drives a pinion **140** meshed with the second gear **136**. The shutter has four blades arranged symmetrically around its axis, with the blades and the gaps between them each subtending 45 degrees at the axis. The blades and the gaps between them are wide enough to block or clear the entire cross-section of the beam, shown in FIG. **13** at **116***a*.

Turning now to FIGS. **3** and **4**, the co-processor **26** is shown providing an eight bit data output to a d/a converter **40** (FIG. **3**) the output of which is amplified by an operational amplifier **41** and supplied to the "COMPEN" terminal of an LM3524 pulse width modulator ic **42** (FIG. **4**). The ic **42** control a P-channel enhancement mode MOSFET Q1 which, when switched on, connects a 24V supply to a motor supply bus **43** through the intermediary of an inductor **44**. The motor is connected in a bridge formed by two push-pull pairs of MOSFETs Q2, Q3 and Q4, Q5. These four MOSFETs are driven by respective driver transistors Q6, Q7, Q8 and Q9. Transistors Q7 and Q9 are respectively controlled by "LEFT" and "RIGHT" outputs taken from the co-processor **26**, so that FETs Q2 and Q5 or FETs Q3 and Q4 are biassed to conduct. Transistors Q6 and Q8 are driven from a 40V supply rail so as to ensure that FETs Q2 and Q4 are turned hard on when conductive, thereby ensuring minimum power dissipation in these devices.

The two FETs Q3 and Q4 are connected to the return bus via a current sensing resister RC, which supplies a current related signal to a voltage comparator **45** with hysteresis to provide an input to the **A6** input terminal of the co-processor

26 when the current exceeds a predetermined limit. This enables the co-processor to reduce the power applied to the motor to maintain it within safe operating limits.

The optical encoder of the pan motor provides two digital outputs in quadrature, these outputs being cleaned up by interface circuits and applied to two inputs of an HCTL-2016 counter ic **46** intended specifically for use with quadrature type encoders. The counter **46** counts up when the pulses are in one relative phase relationship and down when the opposite phase relationship exists. It therefore maintains a count-state related to the motor shaft position and hence the pan angle of the lamp. This count-state is applied to the C0 to C7 terminals of the co-processor **26**. The co-processor **26** also receives "desired value" data from the main lamp cpu **21**, via a 75176 ic **47** (which in fact serves both co-processors **26** and **27**). The ic **47** is used to control the transmission of data between the main lamp cpu and the co-processors. Normally the ic **47** is set to receive data from the cpu **21** and pass it to the two co-processors **26** and **27**. At power-up or when the main lamp cpu **21** transmits a "break" command, the co-processor **26** is reset by a circuit **48**. The co-processor **26** has a cycle time of 1 mS and on receipt of new data it determines the distance to be travelled and then increases the "desired position" value which is compared with the actual position count by one sixteenth of the required change on each successive iteration of its control loop.

The desired value signals passed from the cpu **21** to the co-processor **26** are also time-sliced, being incremented every 16 mS. When new position data is transmitted to the lamp it is accompanied by data representing the length of time over which the movement is to be spread. The data is received, as mentioned above, in the form of two byte numbers respectively representing the x, y and z co-ordinates of a point in a Cartesian co-ordinate system. During initial setting up of the system, each lamp is sent data which informs its cpu **21** of its position in the coordinate system and also of its orientation.

On receipt of a new set of "point at" co-ordinates, the cpu **21** undertakes a "time-slicing" operation to determine how data should be passed to the co-processors **26** and **27**. First of all, it determines how many 16 mS loops will take place in the time duration determined by the data contained in the massage received by the lamp and sets up a variable U equal to the reciprocal of this number. A travel variable P is initialised to zero and the total distance to be travelled is determined for each of the pan and tilt movements. Thereafter, on every iteration of the 16 mS loop the travel variable P is incremented by the reciprocal variable U, the result is multiplied by the total travel required and this is added to (or subtracted from) the previous desired value before transmission to the co-processor **26** or **27**. When the variable P exceeds unity, the target has been reached.

The message sent to the lamp may include a flag indicating whether travel is to occur in a linear fashion as described above or have a sinusoidal profile imposed on it. In the latter case the value of P is modified as follows:

P'=sin (2*P)+0.5*(P>0.5) the latter term being 0 or 1

The main cpu **26** must next convert the x,y,z values into pan and tilt value data for passing to the co-processors **26** and **27**. The cpu first carries out a linear transformation of the absolute x,y,z co-ordinates into co-ordinates x',y',z' relative to the lamp's own frame of reference using the data supplied during initial set up. The ratio of the transformed x' and y' values is calculated as a 16-bit integer, which is used as an index to an ARCTAN table stored in ROM to obtain a value for the desired pan angle. To find the tilt angle, it is first necessary to establish the radial position of the target point in the transformed horizontal plane by calculating the square root of the sum of the squares of the co-ordinates x' and y'. In carrying out this calculation it is necessary to detect an overflow condition which exists if the sum of the squares is a 33 bit number. If this condition is detected, each square is divided by four and a new sum is formed, an overflow flag being set to indicate that overflow has occurred. The square root is found by up to sixteen steps of successive approximation and the result is doubled if the overflow flag was set during the calculation. The resulting square root is divided by the value z' and the result is applied as before to the ARCTAN table to determine the tilt angle. The results obtained represent the new pan and tilt positions to which the lamp is to be moved.

The arrangement described for sending out x, y and z co-ordinate data instead of pan and tilt angle data is highly advantageous in that it enables the console main cpu load to be significantly reduced and also makes it very easy for a console operator to control light beam movements. It is frequently required for a group of lamps to be used together to illuminate a single performer. Where the performer moves from one position on stage to another it is required for all the lamps to change position simultaneously to follow. If the system involved transmission of pan and tilt angle data, this data would be different for every lamp in the group. It would have to be set up by the console operator and stored in cue files on the hard disk drive unit **15**. This would be a very time consuming operation as the pan and tilt angles for each lamp would have to be established and recorded individually. The cue record would need to be of considerable size to record all the different data for each lamp. With the arrangement described above, however, only the x,y,z co-ordinate data needs to be stored and when the cue is recalled the same data is sent to each of the lamps in the group.

Whilst it is theoretically possible to use stored cue data in x,y,z co-ordinate form and to use the console main cpu **14** to calculate the pan and tilt angles to send to the lamps, this would be unsatisfactory as the calculations involved would impose a very heavy load on the cpu **14**, particularly where a large number of lamps in several different groups had to be moved as the result of a single cue.

As described above a "point-at" mode is envisaged as the normal operating mode. However, other modes of operation are also envisaged. For example, the lamp could be instructed to point away from the point specified or to point in a direction parallel to a line joining a fixed point (eg the origin of the co-ordinate system) to the point specified. These "point-away" and "point parallel" modes would be selected by means of flags included in the data transmitted to the lamps.

The arrangement described enables the lamps to be very precisely synchronised. The data is transmitted from the distribution unit to all of the lamps simultaneously and each lamp can start to respond at the end of the message. This enables very precise direction of all the lamps to a moving point in "point-at" mode and very clean parallel sweeps to be made in "point parallel" mode.

It should be noted that the use of x,y,z co-ordinates is also very advantageous in situations where a pre-arranged lighting performance is to be used in several different venues. The pre-loaded gantries or trusses used for such touring performances cannot always be mounted at exactly the required positions relative to the stage because of local conditions. In this case all that is needed is for offsets data

to be sent to the lamps at set-up time to enable each lamp cpu to correct its position data. No editing of the individual pre-recorded cues is necessary as it would be in the same circumstances if pan and tilt data were stored.

As part of the set-up procedure for each performance it is necessary to initialise the values of the actual pan and tilt angle count-states, since encoders of the type used do not give any absolute position data. This is accomplished by driving the lamp to an end stop in one direction for each movement. The lamp is driven back to a predetermined number of counts and the counters are reset to zero at this position.

Turning now to FIGS. **5** to **7**, the circuitry for controlling the individual dc servo-motors inside the lamp is more complex as each co-processor has to deal with up to six servo-motors. As shown in FIG. **5**, the co-processor **28** controls a number of data routers **50** to **54** which determine which channel is being controlled at any given time. The router **50** co-operates with six HCTL-2016 counters **55** which count the quadrature pulse outputs of the respective encoders, to determine which of the counters should supply its count-state to the co-processor **28**. Router **51** controls individual resetting of the counters **55**. Router **52** co-operates with a 74HC175 ic **56** (one for each channel) to determine which L6202 ic motor controller **57** is enabled and also routes "RIGHT" and "LEFT" signals from the co-processor to the circuits **57**. Router **53** controls routing of position error data calculated by the co-processor **28** for each channel to latches **58** (one for each channel) at the input of pulse width modulator circuits for controlling the motor controllers **57**. This error data is actually passed to the latch **58** in an inverted form, so that the larger the error, the smaller the value passed is. Router **54** routes various digital sensor signals to a sensor input of the co-processor, Such sensors are utilized by some of the channels to indicate when the moving part in question is in a datum position. This is required for the gobo wheels, the colour wheels and the shutter, but not for the iris diaphragms or lenses which can be moved to end stop positions. During datum set-up the sensors (optical sensors sensing a hole or flag or Hall effect sensors) are detected and the HCTL counters are reset.

As co-processor **28** has only 256 bytes of internal memory, extra memory is required for each channel to store program variables. The RAM selection control circuit is shown in FIG. **7**. The memory ic **59** (an HM6116LP ic) has 11 address lines of which eight are connected to the co-processor write bus via a latch circuit **60** and the remaining three of which are connected to spoare outputs of three of the ics **56**. Spare outputs of the selectors **50, 51, 52** are connected to control terminals of the memory ic and a spare output of the selector **53** is connected to an output enable terminal of the latch circuit **59**. Thus a particular address in the memory ic can be selected by the co-processor by first setting the ics **56** and the selectors **50, 51, 52** to appropriate states and then outputting the lower bytes of the address to latch **60** whilst output from latch **60** is enabled. Two further eight bit latches **61** and **62** provide temporary storage for data to be written to and data just read from the memory ic **59**. When neither reads nor writes are required the memory data bus is tri-stated. Bus contention is thus avoided.

Circuit **57** actually controls the motor current, but it in turn is controlled by a pulse width modulator circuit, comprising the latch **58** and a digital comparator **65** which compares the contents of latch **58** with the count-state of an 8-bit continuously running counter **66***a*, **66***b* serving all channels. The comparator output goes high when the count-state exceeds the latch contents, so that if the latch content is low the comparator output is high for a high proportion of each cycle of the counter **66***a*, **66***b*. The output of the comparator **65** is ANDed with an enable output from ic **56** by a gate **67** and then with the output of an overcurrent detector circuit **68** by another gate **69**.

When a new target value for one of the parameters controlled by co-processor **58** arrives in the receive buffer, and it is associated with execution duration data (this may apply to lens movements, colour changer movements, gobo movements and iris diaphragm movements, but not shutter movements) the cpu **21** handles time slicing as in the pan and tilt operations. Since several channels are controlled by each co-processor, however, no interpolation by the co-processor is used. Instead each channel has its error checked and a new value written (if necessary) to latch **58** every 12 mS

In the case of the shutter, the message received by the lamp merely includes a shutter open or shutter closed command. When the required shutter status changes, the main cpu merely increases the target shutter angle by 45 degrees (in the case of a four bladed shutter) and passes the new value to the co-processor.

This arrangement enables the shutters of some or all of the lamps to be operated in synchronism. Moreover, the console cpu **14**, can operate to update the shutter open/closed instructions at regular intervals to obtain a stroboscopic effect, synchronised for all the lights.

We claim:

1. A lighting control apparatus which controls a positioning of a beam of a lamp, comprising:

a main control console accepting user input relating to required beam movements;

at least two independently operable lamp units, each situated in a different location, and each situated remotely from the main control console, wherein each of the lamp units includes a servo-mechanism which operates to automatically move the lamp beam about two mutually transverse axes to a desired angular position responsive to an applied command;

a data communication element coupled to the main control console and to the lamp units and operating to transmit desired position data to said at least two lamp units, wherein the desired position data is transmitted in the form of a set of three-dimensional absolute linear coordinates, defining a point in space through which the lamp beam is required to pass and each of said at least two lamp units receiving the same absolute linear coordinates; and

a calculating device, located in each lamp unit, receiving the absolute linear coordinates and calculating a desired angular position based on relative coordinates and supplying the servo-mechanism with the desired angular position.

2. The lighting control apparatus of claim **1**, wherein the desired position data received by the lamp units also includes data defining a time duration over which execution of the movement of the lamp beam to its new position is specified by the desired position data, and said calculating device divides execution of a position change into a number of steps, where the number is inversely related to the execution duration for re-calculating the desired position data at intervals, and subjects the re-calculated data to calculate the desired angular position data for each step.

3. The lighting control apparatus of claim **1**, wherein said calculating device in each lamp unit comprises a lamp main cpu programmed to carry out the absolute linear coordinate to angle calculations.

4. The lighting control apparatus of claim 1, wherein said main control console includes a disk drive unit storing set-up data and cue data, said set-up data being transmitted to the lamp units during setting up of the apparatus, and the set-up data being supplemented, when required, by offset data indicating a displacement of the lamp units from normal expected positions relative to a target area.

5. An apparatus as in claim 1 wherein said calculating device includes an element which linearly transforms the absolute coordinates to the relative coordinates that are relative to the lamp's own frame of reference.

6. An apparatus as in claim 5 wherein the calculating device includes a memory storing a ratio between values, and wherein said calculating device uses said ratio to obtain a value for desired angle.

7. An apparatus as in claim 1 wherein said absolute coordinate system is an x, y, and z coordinate system, and said calculating device includes an element carrying out a transformation of the absolute x, y, and z coordinates into coordinate x'y'z' which are relative to the lamp's own frame of reference.

8. A lighting control apparatus, comprising:

a main control console, accepting user input indicating a desired position for a light beam, and producing an output signal indicative of a desired absolute position of the light beam relative to a frame of reference stored in the console;

at least two lamp units, each of said at least two lamp units located at different locations, and each connected to said console to receive a message therefrom, each said message including an indication of said absolute position of said lamp unit within the frame of reference stored in said main console, each of said at least two lamp units including a calculating element, receiving information indicating said position of said lamp unit in said absolute coordinate system, and converting said information into a value which can be used to adjust a position of said lamp unit.

9. An apparatus as in claim 8 wherein said value includes pan and tilt of the lamp unit to provide a spot from the lamp unit to said desired position.

10. An apparatus as in claim 8 wherein said absolute coordinate system is an x, y, and z coordinate system, and said calculating device includes an element carrying out a transformation of the absolute x, y, and z coordinates into coordinate x'y'z' which are relative to the lamp's own frame of reference.

11. A method of operating multiple lamps in a multiple lamp system, comprising:

determining a desired position of said lamp units in an absolute coordinate system;

sending an indication of said position in said absolute coordinate system to said lamp units;

calculating, at each of said lamp units, a translation between said absolute coordinate system and a local coordinate system which is local to said each lamp unit; and

moving said lamp unit to a position indicated by said local coordinate system.

12. A method of operating multiple lamps in a multiple lamp automatic system, comprising:

establishing a plurality of prerecorded cues, each said cue including information about positions of more than one lamp;

executing a cue by sending information to said more than one lamp in an absolute coordinate system, said information in said absolute coordinate system being a representation which is stored in said console, and each of said items of information being sent to each of said lamps in said absolute coordinate system;

calculating, in each lamp, a conversion between said absolute coordinate system and a relative coordinate system of said each lamp; and

carrying out an operation to move the lamp to a position indicated by said absolute coordinates system, so that said cue is executed separately by each lamp based on the common data sent to each lamp.

13. A method as in claim 12, wherein said carrying out comprises using a prestored ratio to convert from said absolute coordinate system to said relative coordinate system.

* * * * *

EXHIBIT D

US005921659A

United States Patent [19]

Hunt et al.

[11] Patent Number: 5,921,659

[45] Date of Patent: Jul. 13, 1999

[54] STAGE LIGHTING LAMP UNIT AND STAGE LIGHTING SYSTEM INCLUDING SUCH UNIT

[75] Inventors: **Mark A. Hunt**, Derby; **Keith J. Owen**, Moseley; **Michael D. Hughes**, Wolverhampton, all of United Kingdom

[73] Assignee: **Light & Sound Design, Ltd.**, Birmingham, United Kingdom

[21] Appl. No.: **08/994,036**

[22] Filed: **Dec. 18, 1997**

Related U.S. Application Data

[62] Division of application No. 08/576,211, Dec. 21, 1995, Pat. No. 5,788,365, which is a continuation of application No. 08/077,877, Jun. 18, 1993, Pat. No. 5,502,627.

[51] **Int. Cl.**[6] **F21S 1/14**

[52] **U.S. Cl.** **362/233**; 362/272; 362/286

[58] **Field of Search** 362/233, 269, 362/272, 286

[56] **References Cited**

U.S. PATENT DOCUMENTS

4,392,187	7/1983	Bornhorst	362/233
4,980,806	12/1990	Taylor et al.	362/233 X
5,045,983	9/1991	Shields	367/293
5,502,627	3/1996	Hunt et al.	362/286
5,769,531	6/1998	Hunt et al.	362/286 X
5,788,365	8/1998	Hunt et al.	362/301

FOREIGN PATENT DOCUMENTS

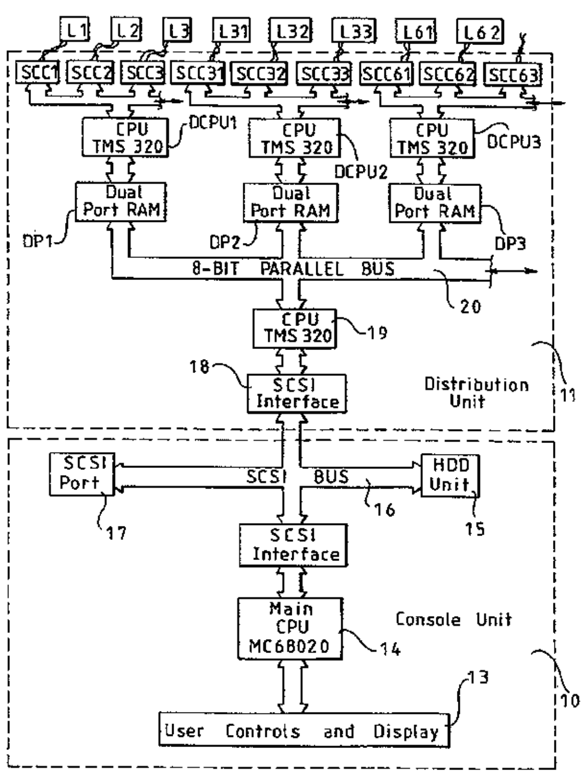
0 495 305	7/1992	European Pat. Off. .
2 643 134	8/1990	France .
2 209 229	5/1989	United Kingdom .
2 231 138	11/1990	United Kingdom .
89 054 21	6/1989	WIPO .

Primary Examiner—Stephen Husar
Attorney, Agent, or Firm—Fish & Richardson P.C.

[57] **ABSTRACT**

A stage lighting lamp unit includes a processor for receiving control data from a remote console. Beam orientation data for the lamp unit is passed to the lamp in the form of the x, y and z coordinates of a point in space through which the beam is to pass. The processor divides the required lamp travel into a number of stages dependent on execution duration data sent with the position data, and calculates, for each stage, a new value for pan and tilt angles for the lamp. These values are passed to pan and tilt controlling co-processors which control servo-motors for pan and tilt operation. The lamp unit also incorporates a rotatable shutter for interrupting the lamp beam when required. The shutters of all the lamps in a system can be instructed from the remote console to open and close in synchronism, thereby providing a stroboscopic effect.

11 Claims, 10 Drawing Sheets



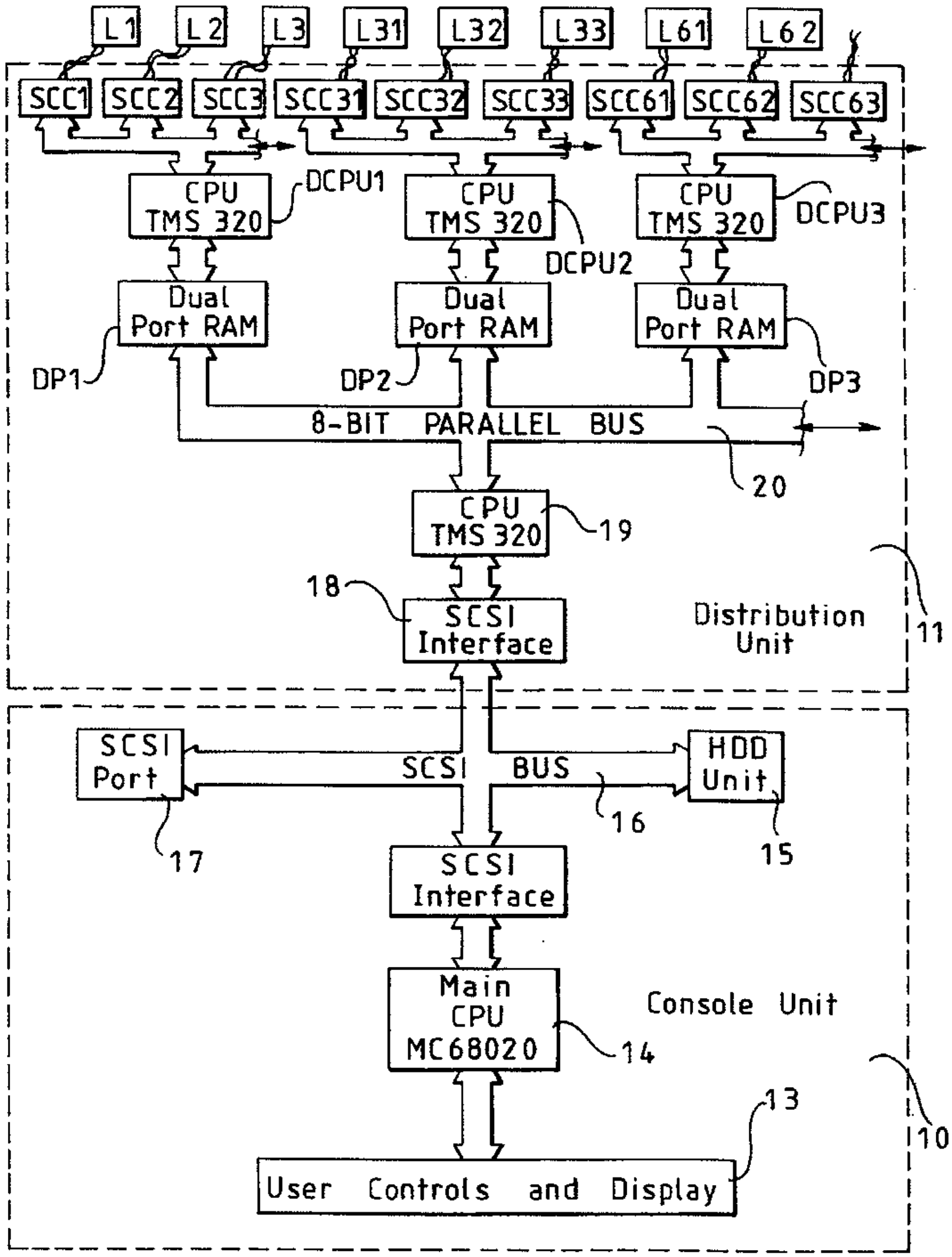

L1
L2
L3
L31
L32
L33
L61
L62
SCC1
SCC2
SCC3
SCC31
SCC32
SCC33
SCC61
SCC62
SCC63
CPU TMS 320
DCPU1
CPU TMS 320
DCPU2
CPU TMS 320
DCPU3
Dual Port RAM
Dual Port RAM
Dual Port RAM
DP1
DP2
DP3
8-BIT PARALLEL BUS
20
CPU TMS 320
19
18
SCSI Interface
Distribution Unit
11
SCSI Port
SCSI BUS
HDD Unit
16
17
15
SCSI Interface
Main CPU MC68020
Console Unit
14
13
10
User Controls and Display

FIG 1

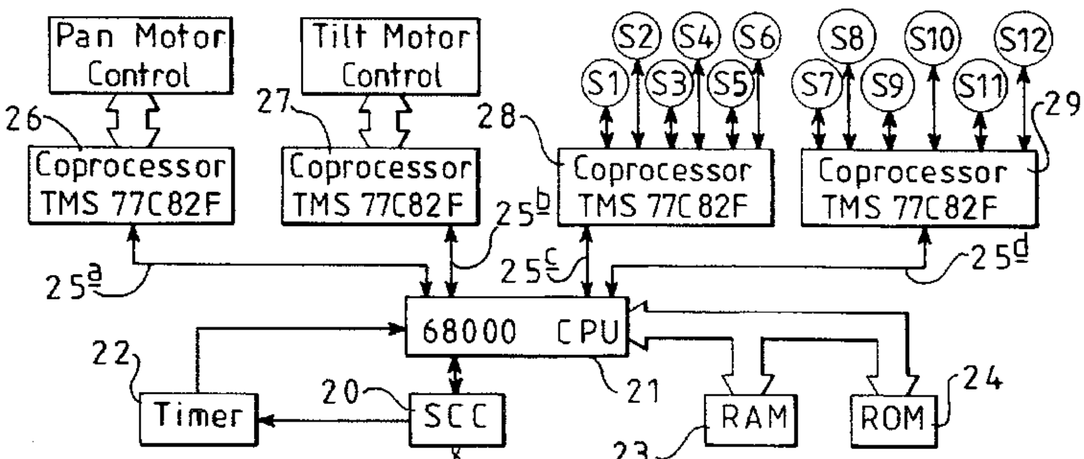

Pan Motor Control
Tilt Motor Control
S1
S2
S3
S4
S5
S6
S7
S8
S9
S10
S11
S12
29
26
27
28
Coprocessor TMS 77C82F
Coprocessor TMS 77C82F
Coprocessor TMS 77C82F
Coprocessor TMS 77C82F
25b
25a
25c
25d
68000 CPU
22
20
21
24
Timer
SCC
RAM
ROM
23

FIG 2

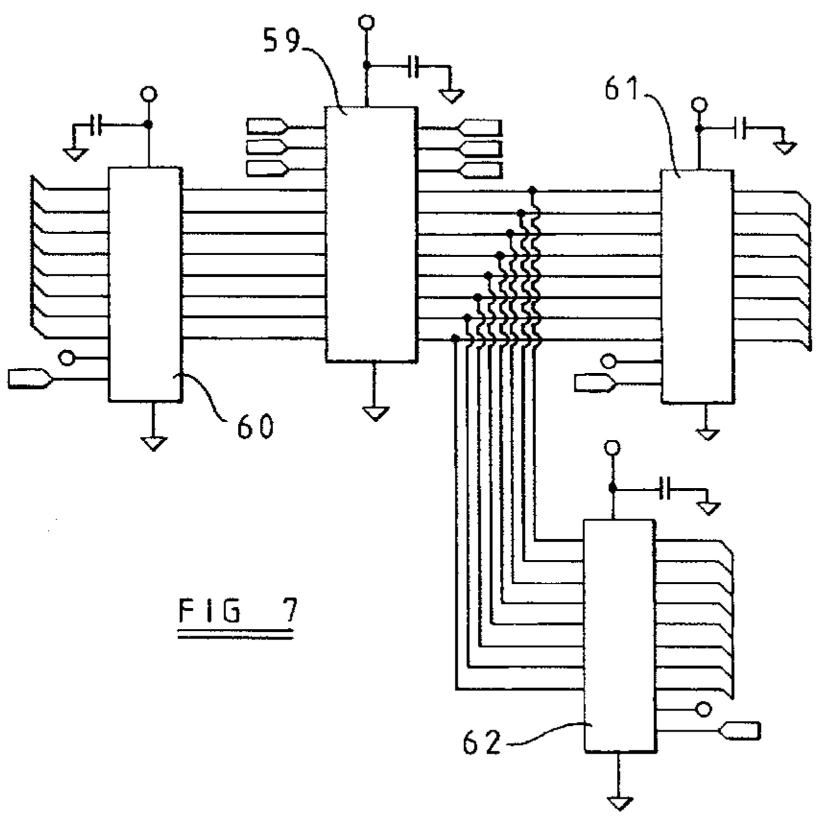

59
61
60
62

FIG 7

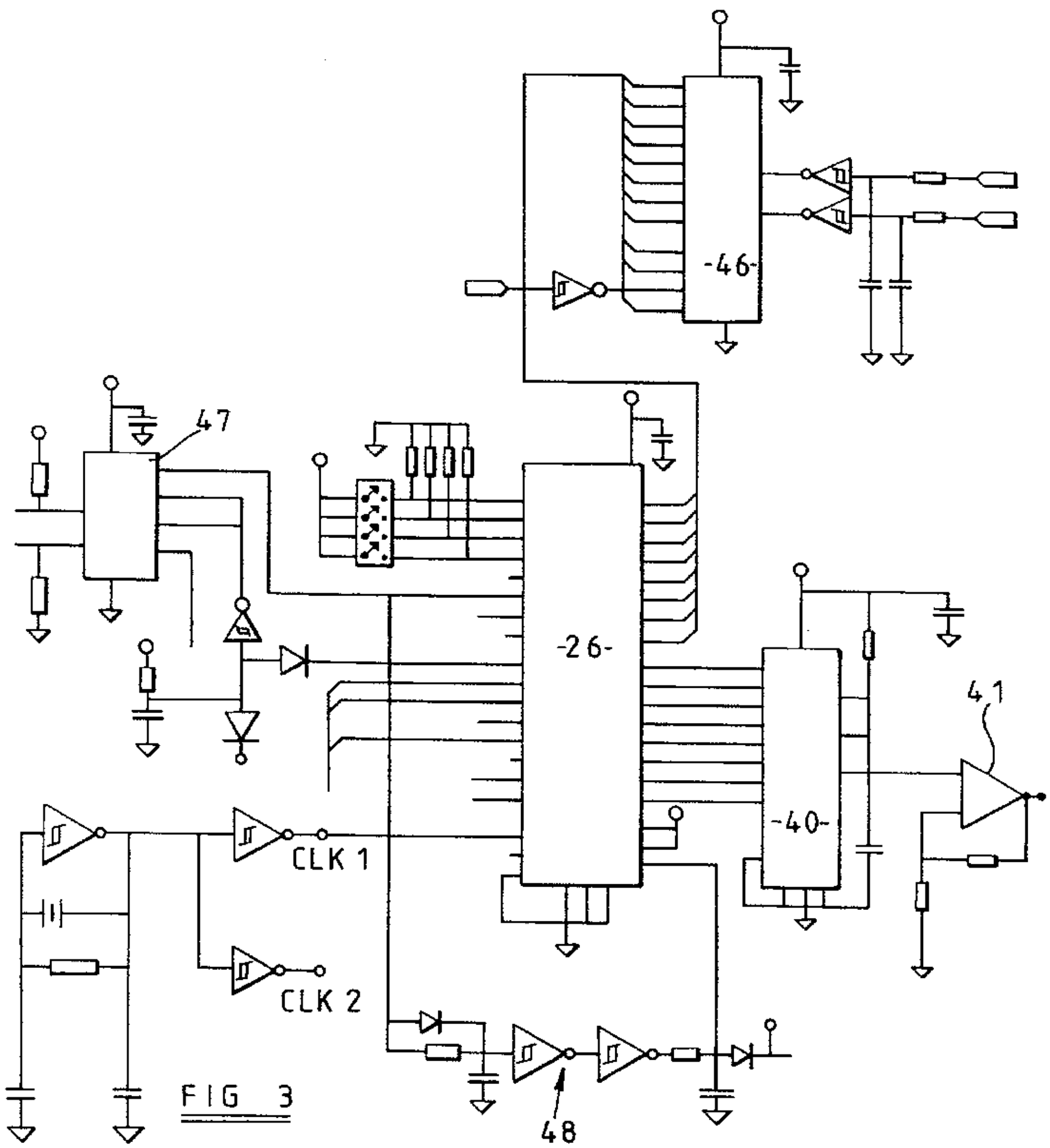
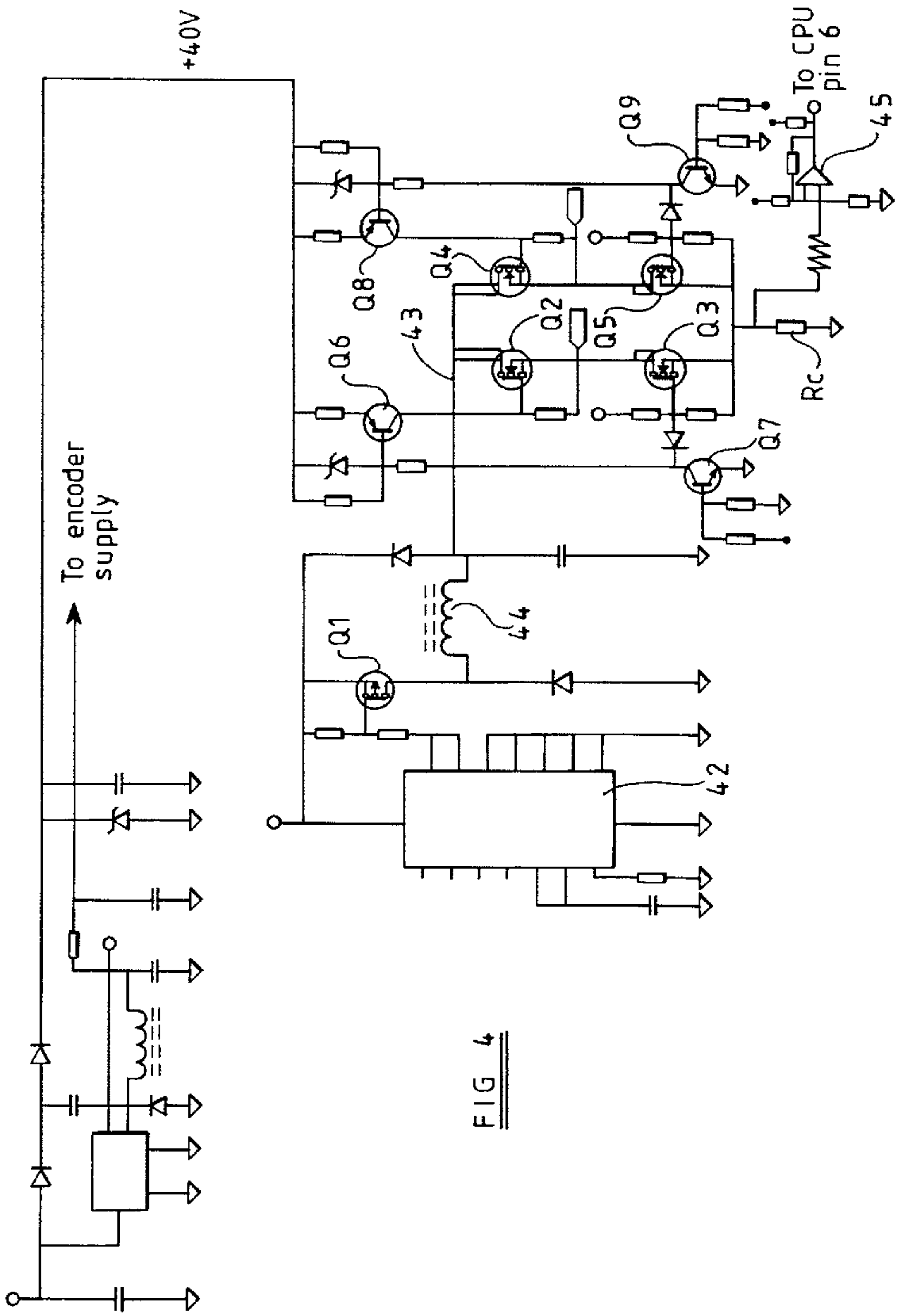


FIG 3



+40V
To CPU pin 6
45
Q9
Q8
Q4
Q2
Q5
Q3
Q6
43
Rc
Q7
To encoder supply
Q1
44
42

FIG 4

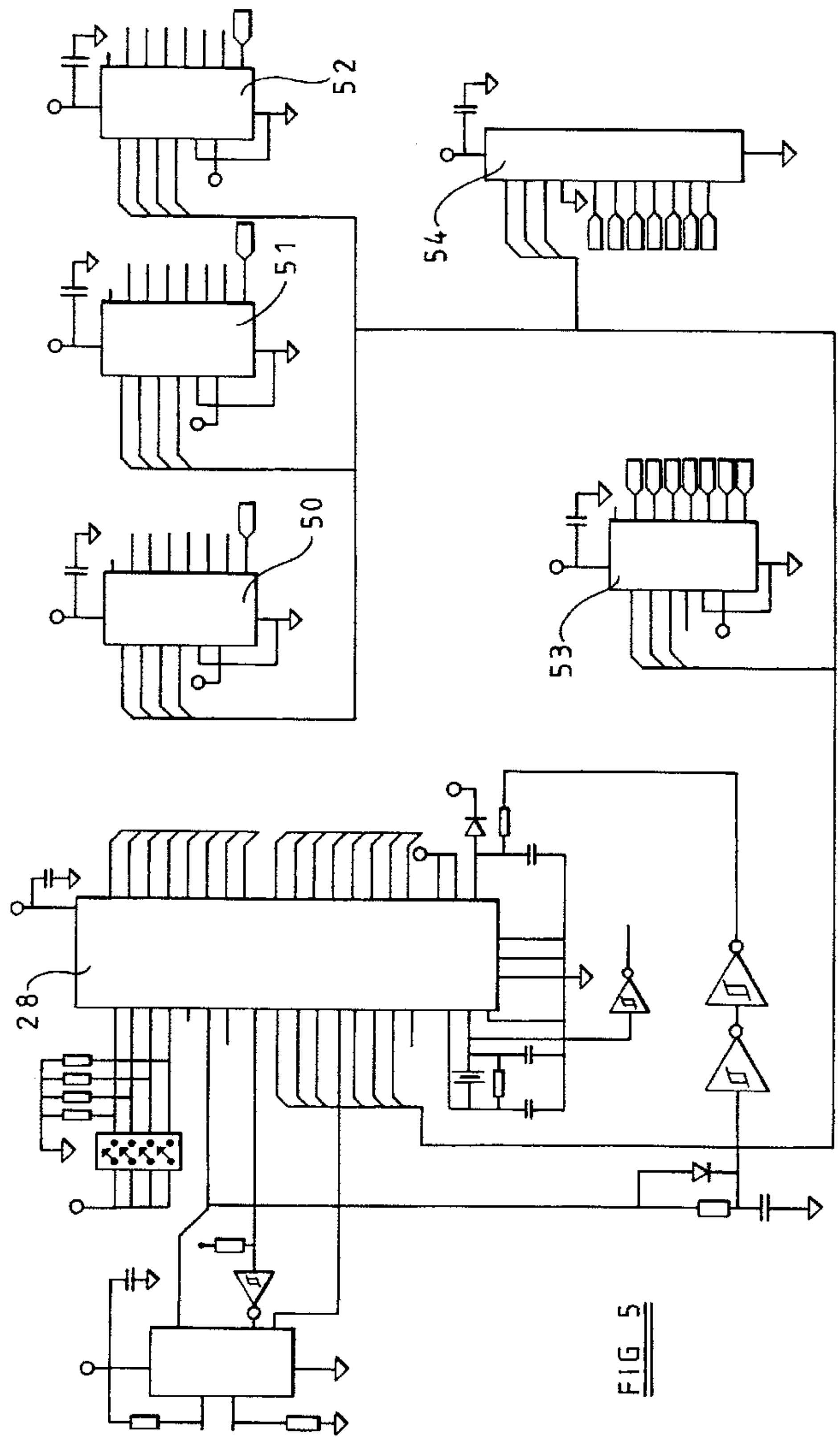


FIG 5

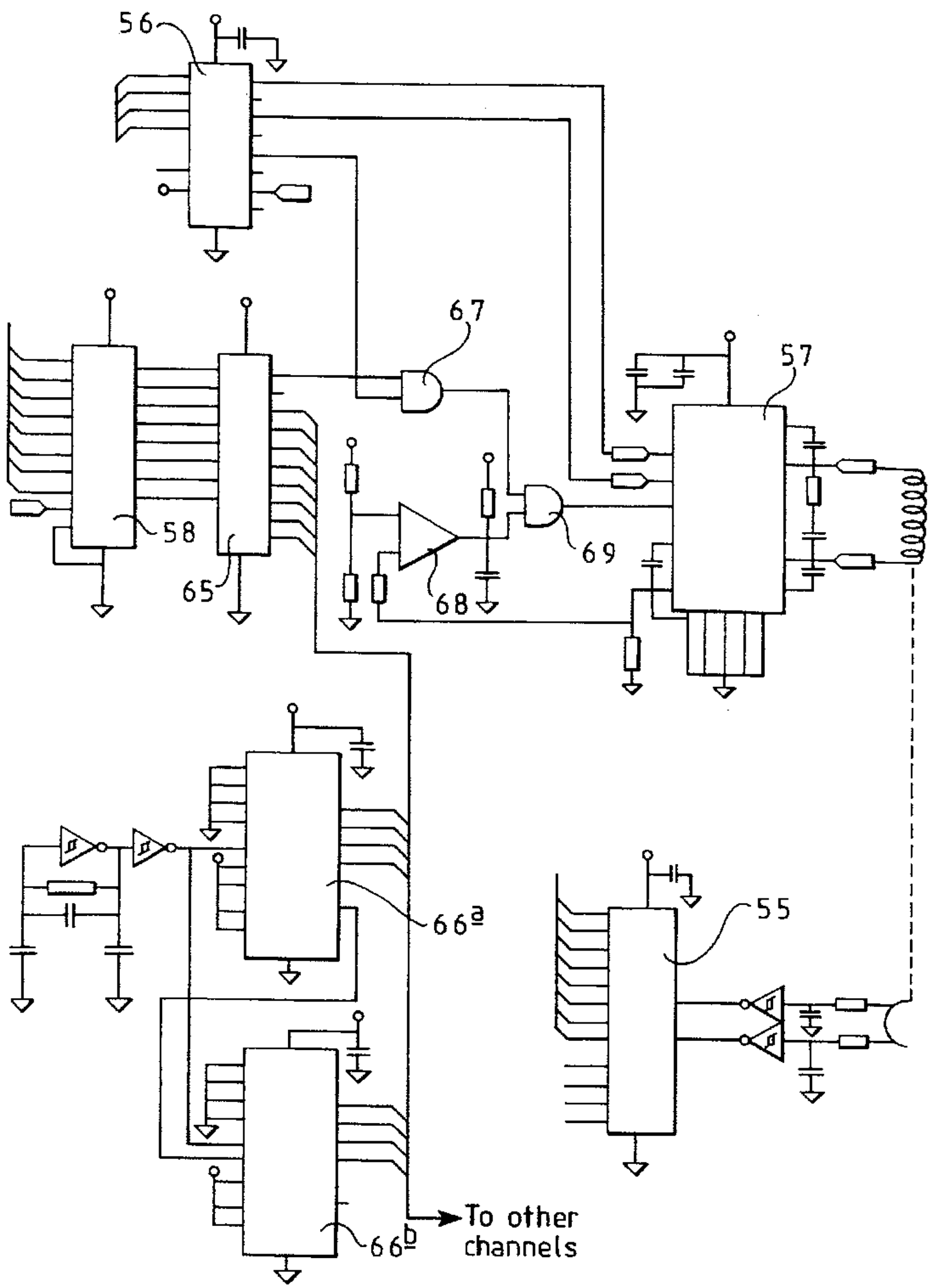


FIG 6

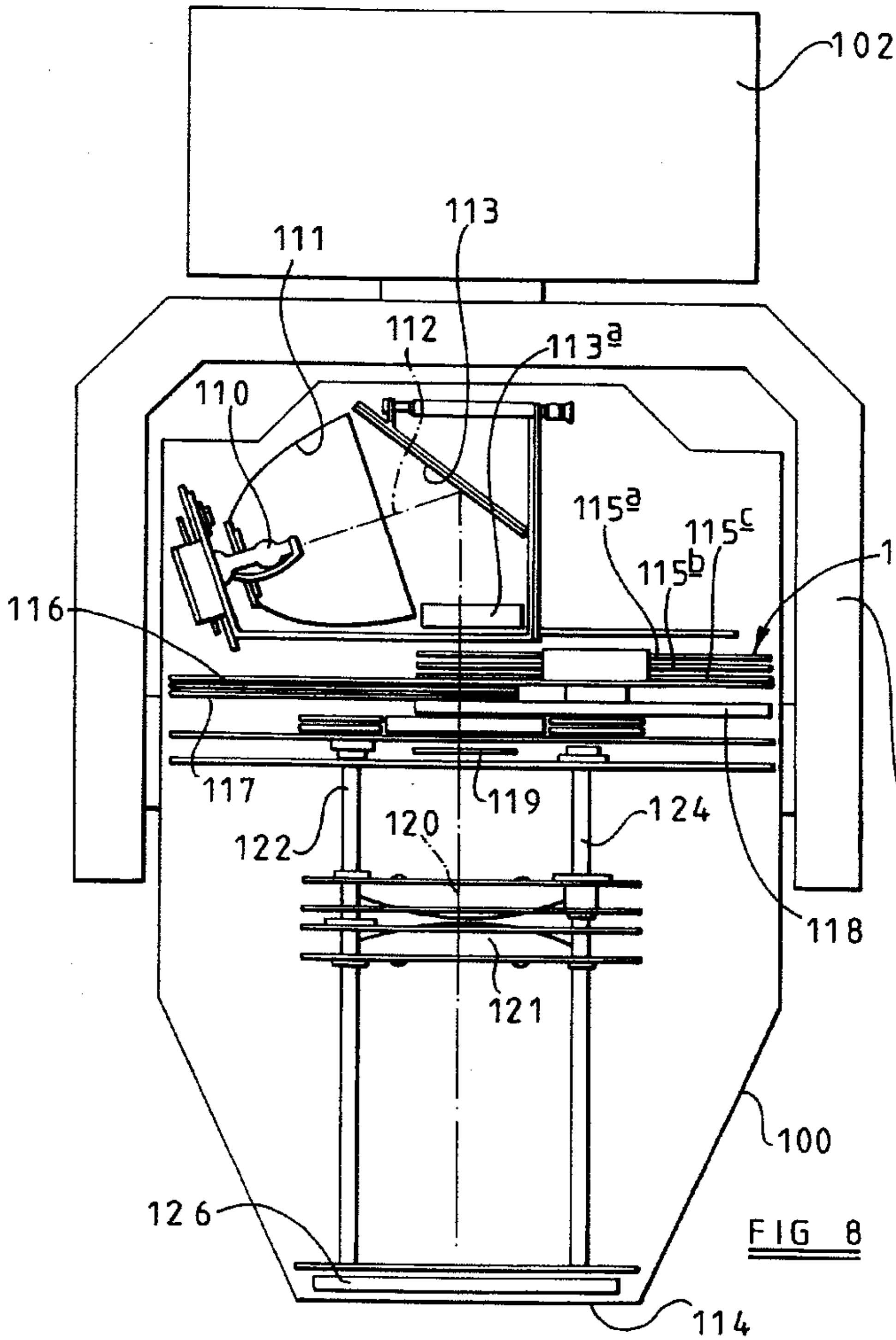

102
113
111
112
113a
110
115a
115c
115b
115
116
101
117
119
124
122
120
118
121
100
126
114

FIG 8

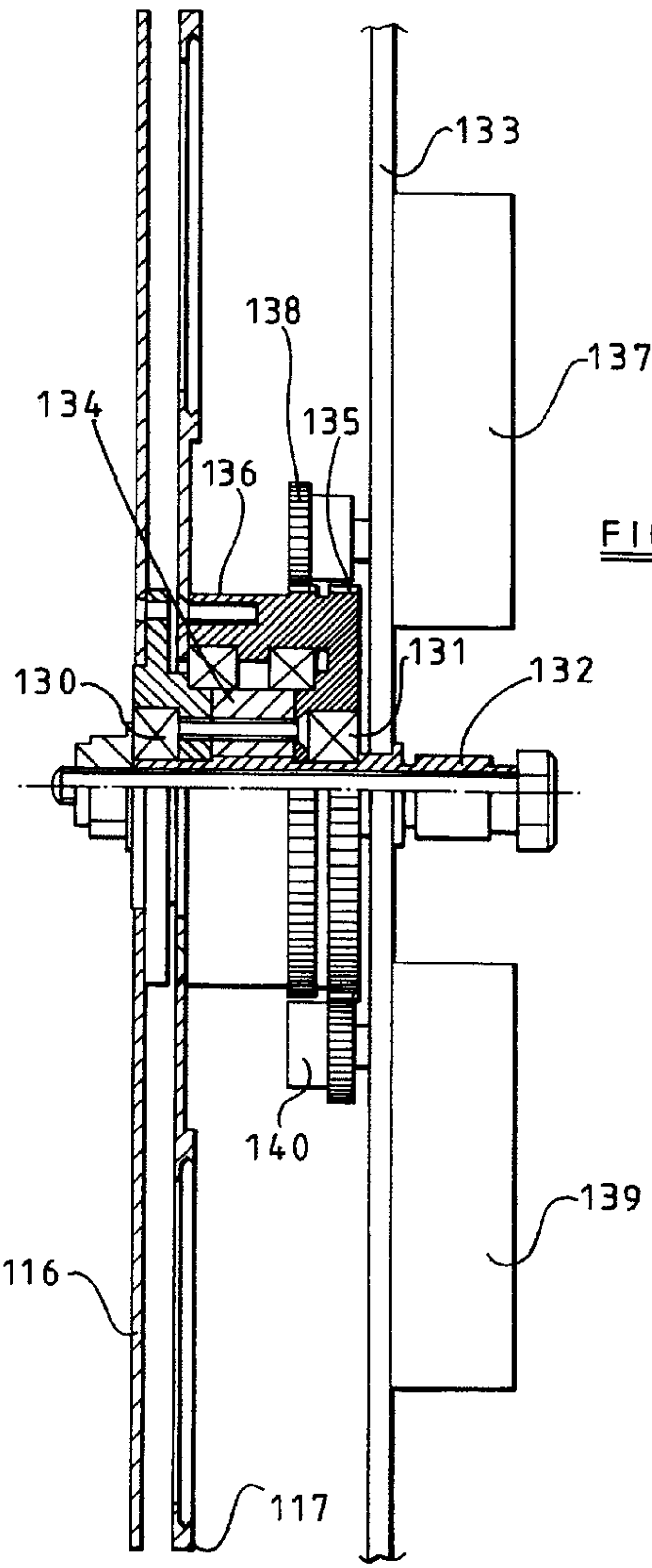
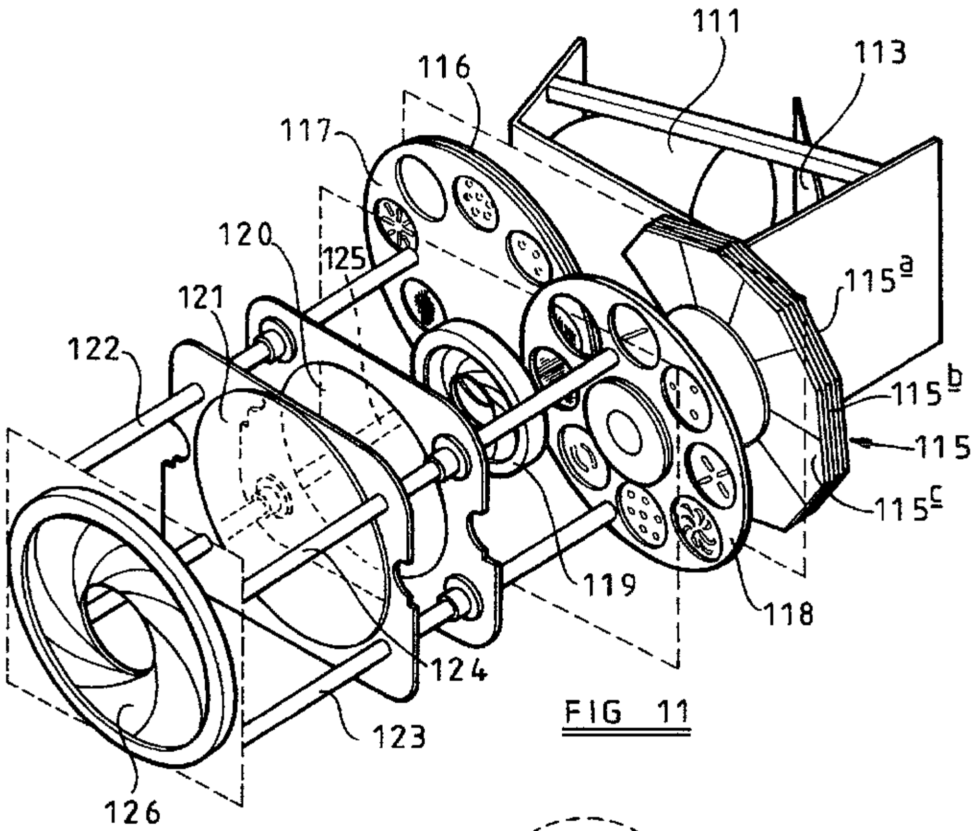
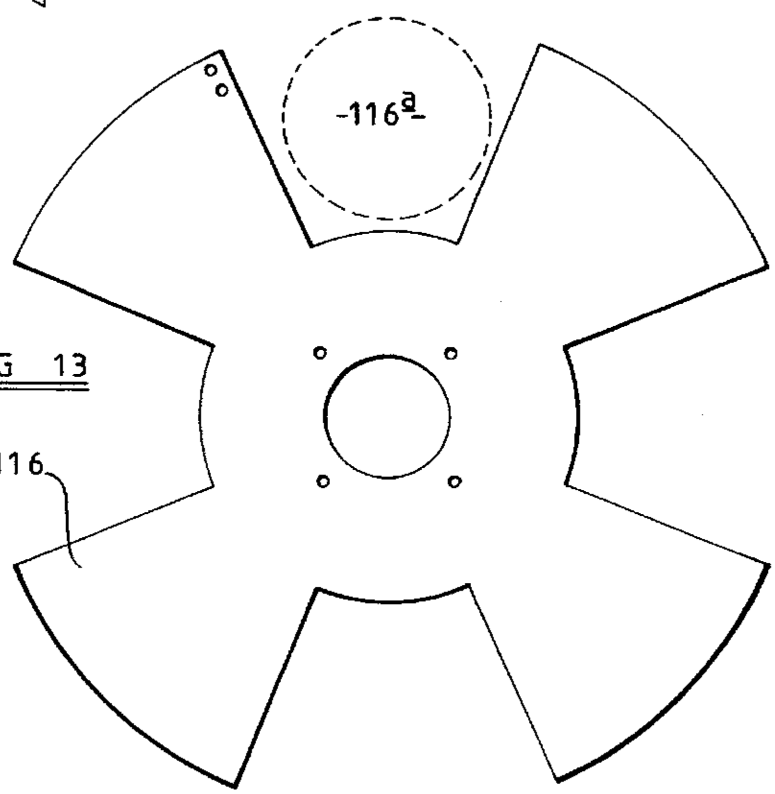


FIG 12



111
113
116
117
120
125
121
122
115a
115b
115
115c
119
118
124
123
126

FIG 11



116a
116

FIG 13

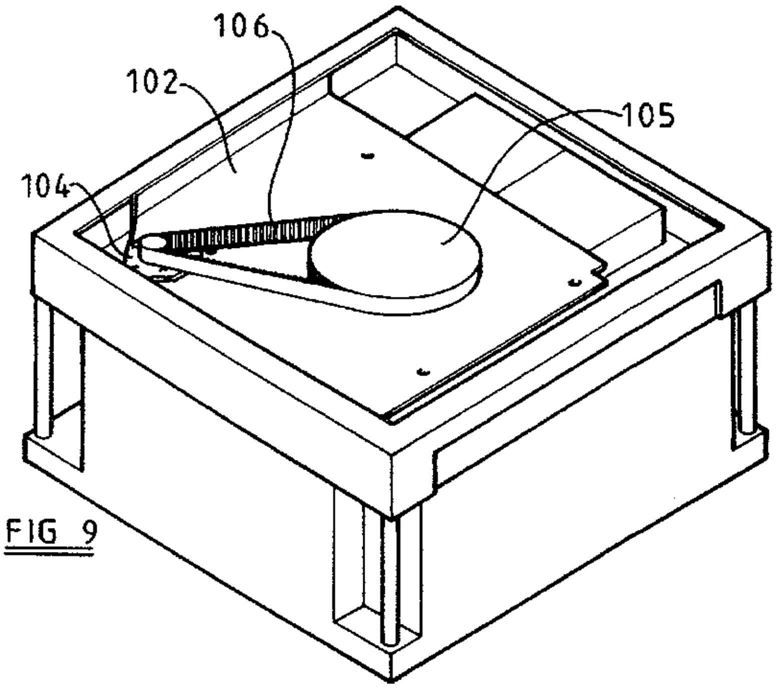


FIG 9

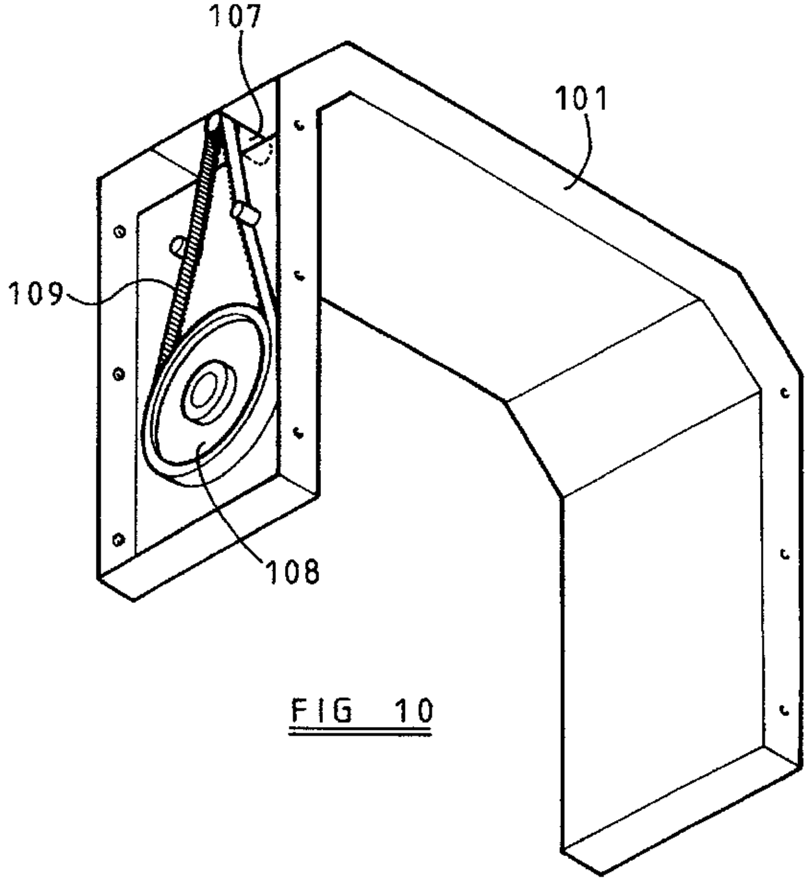


FIG 10

STAGE LIGHTING LAMP UNIT AND STAGE LIGHTING SYSTEM INCLUDING SUCH UNIT

This is a divisional of U.S. application Ser. No. 08/576,211, filed Dec. 21, 1995, now U.S. Pat. No. 5,788,365, which was a continuation of Ser. No. 08/077,877 filed Jun. 18, 1993 now U.S. Pat. No. 5,502,627.

This invention relates to stage lighting and is particularly concerned with the control of multiple functions of a lamp.

It has already been proposed to incorporate in a lamp unit a plurality of different functions, such as colour changers, focusing lenses, iris diaphragms, gobo selectors and pan and tilt mechanisms which are controlled from a remote console. Stage lighting systems have as a result reached very high levels of complexity requiring a very complicated main control console and lamp unit constructions. The use of microprocessors, both in the console and the lamps has become conventional as increasing complexity makes it more difficult to produce and subsequently maintain a system which uses hard wired logic or analog controls. In such systems the microprocessor in the console is used to allow the user to set up lighting cues and to control the sending of appropriate data to the lamp microprocessors. The lamp microprocessors are also involved in controlling communication between the console and the lamps, and also have to control a plurality of servo-motors which drive the various functions of the lamps.

It is one object of the present invention to provide a lamp micro-processor and servo-control arrangement which allows complex functions to be carried out.

It is another object of the invention to provide a lamp control system in which control of pan and tilt movements of each lamp can be carried out in rapid and efficient manner, enabling large groups of lamps to make coordinated movements.

It is yet another object of the invention to provide each lamp in a stage lighting system with a means for quickly interrupting its light beam and quickly re-stablishing the beam so that a group of lamps can be made, when required to flash in synchronism.

In accordance with one aspect of the invention there is provided a lamp unit for connection to a remote control console for the control of a plurality of different functions of the lamp, said unit comprising a main processor circuit, associated with a communication controller for accepting message data from the console, a plurality of servo-controls for operating said functions of the lamp, and a plurality of co-processors which are connected to the main processor circuit so as to be supplied thereby with desired value data for the various lamp functions, said servo-controls being controlled by said co-processors.

In the case of pan and tilt controls where close control is required throughout the movement of the lamp from an initial position to a new position, one of the co-processors is assigned solely to the control of movement about each axis. Other functions can share a co-processor.

The main processor circuit of the lamp is preferably programmed to accept data from the control console defining not only a target position for any function, but also a duration over which the function is to be executed. In this case the main processor circuit divides the "journey" into segments and updates the target position data passed to the associated co-processor at intervals.

In accordance with another aspect of the invention, there is provided a lighting control apparatus comprising the combination of a main control console for accepting user

input relating to required beam movements, a plurality of independently operable lamp units situated remotely from the console, each of the lamp units incorporating a servo-mechanism for automatically moving the lamp beam about two mutually transverse axes to a desired angular position and data communication means connecting the console to the lamp units for the transmission of desired position data to the lamp units, the desired position data being transmitted in the form of a set of three dimensional linear co-ordinates defining a point in space through which the lamp beam is required to pass, and each lamp unit including a calculating device for calculating the desired angular position from the desired position data and supplying the servo-mechanism with such desired angular position.

In addition to the "point at" mode of operation mentioned above, additional modes may be specified in which the lamps point away from the specified point or in which they all point in the same direction parallel to a line between a fixed position in the co-ordinate system and the specified point.

Conveniently, all the data concerning the positions and orientations of the individual lamp units within the co-ordinate system is stored in a set-up file kept on a hard disk drive in the console. When the same lighting set-up is used at different venues, where it is impossible to set the frame which carries all the lamp units at exactly the same position as that for which the set-up was designed, offset data can be input at the console and either used within the console microcomputer to correct the position data stored during set-up as it is sent out, or such data can be sent to all of the lamp units over the network and stored there, to enable the corrections to be made in the individual lamp processor units.

In accordance with another aspect of the invention, a stage lighting unit comprises a housing, a light source within said housing, an optical system for forming light from said light source into a beam, a rotary shutter device having a plurality of blades, said shutter device being rotatably mounted in the housing so as to cause said blades to pass through and obstruct said beam as the shutter device rotates, a motor for rotating said shutter device and a servo-control for controlling said motor in accordance with data received in use from a remote control console.

The invention also resides in a stage lighting system incorporating a plurality of lighting units as defined above controlled by a common remote control console via data communication means, whereby the rotary shutter devices of all the units can operate in synchronism.

An example of the invention will now be described with reference to the accompanying drawings, in which:

FIG. **1** is a block diagram of a stage lighting system;

FIG. **2** is a block diagram of the internal circuitry of one of a plurality of lamp units in the system of FIG. **1**;

FIGS. **3** and **4** are more detailed circuit diagrams showing a pan motor drive control forming part of the internal circuitry of the lamp;

FIGS. **4** to **7** are detailed circuit diagrams showing a rotary shutter motor drive control forming part of the internal circuitry of the lamp;

FIG. **8** is a diagrammatic, part-sectional view of one of the lamps;

FIG. **9** is a perspective view of a pan movement drive arrangement;

FIG. **10** is a perspective view of a tilt. movement drive arrangement;

FIG. **11** is a diagrammatic perspective view of the internal moving parts of the lamp;

FIG. 12 is a sectional view showing the drive arrangement for a shutter and a gobo wheel forming part of the lamp; and

FIG. 13 is an elevation of a shutter wheel forming part of the lamp.

Referring firstly to FIG. 1, the system consists basically of a console unit 10, a signal distribution unit 11 and a plurality of lamps L1, L2, L3 . . . , L31, L32, L33 . . . , L61, L62 . . . individually connected by twisted pair data communication links to the distribution unit.

The console unit 10 has an array of switches, slider potentiometers, rotary digital encoders and other user actuable input devices (not shown) and a display indicated at 13. These are all connected to main console cpu 14 (an MC68020 microprocessor) which has the task of receiving inputs from the user actuable input devices and controlling the display. Both tasks are assisted by separate co-processors which directly interface with different parts of the console.

The main cpu can communicate with a hard disk drive unit 15 via a SCSI bus 16 which also connects it to the distribution unit and to an external SCSI port 17, through the intermediary of which the console can, if required be connected to a personal computer. The user controls can be used in setting up a sequence of cues in advance of a performance, the sequence being stored in a cue file on the hard disk drive unit 15. The sequence can be recalled during the performance to enable the various stored cues to be executed. Direct manual control of the lamps from the console is also possible as is manual editing of cues called up from the hard disk. The main console cpu 14 creates messages to be sent to the individual lamps, each message comprising a fixed number of bytes for each lamp. The messages contain data relating to the required lamp orientation, beam coloration, iris diaphragm diameter, gobo selection and rotation, zoom projection lens control and opening or closing of a shutter included in the lamp. A block of the RAM of the main cpu is set aside for the storage of these messages, the block being large enough to contain messages for 240 lamps, being the largest number which can be controlled via the distribution unit. Where it is required to control more than 240 lamps additional distribution units can be connected to the SCSI bus and extra main cpu RAM reserved for message storage. When any message data is changed the main cpu 14 sets a flag in the RAM block which is detected at a given point in the main cpu program loop and interpreted as a signal that the changed message data is to be transferred to the distribution unit 11.

The distribution unit 11 has a main cpu 19 which controls reception of data from the SCSI bus interface and distribution of such data to up to eight blocks of dual,port memory DP1, DP2, DP3 . . . via an eight bit data bus 20. The cpu 19 is alerted to the waiting message data when cpu 14 selects the distribution unit. The cpu 19 then supervises byte by byte transfer of the message data which it routes to the various blocks of dual port memory.

For actually sending out the message data to the lamps, there are a plurality of serial communication controllers SCC1 to SCC30, SCC31 to SCC60 etc, there being thirty serial communication controllers associated with each block of dual port memory. A further cpu DCPU1, DCPU2, etc is associated with each block of dual port memory and distributes message data transferred to the dual port memory to the individual serial communication controllers and the messages are transferred to the lamps. Each serial communication controller in the distribution unit includes a line driver which can be disabled except when data is to be transmitted. Enabling of the driver can cause a spurious

signal to be transmitted over the data link. To allow such spurious signals to be identified and ignored, a two-byte gap is left between enabling the line driver and commencing transmission of the message data for the channel in question.

This will be described in more detail herein. All asynchronous serial communication systems require framing information to synchronize the reception process. This has been typically done in the prior art using start bits and stop bits.

The present invention preferably uses FM0 coding in which the data is transmitted as one cycle of the carrier frequency for a zero or as a half cycle of the carrier frequency for a one. When the line has been idle, no waveform at all is present. When the line drivers are first enabled, an arbitrarily short pulse will usually appear on the line, due to lack of synchronization between the data signal and the enabling signal. This short data pulse could be misinterpreted as a start bit, for example and if so it would disturb later framing.

The present invention avoids any problems from this arbitrarily short pulse. To avoid this, the present invention uses a timer on the receive line, set to the time needed to receive two bytes on the serial data line. This timer is restarted whenever a byte on the data line is detected.

Each time the timer interrupt occurs, the number of bytes received is checked against the number of bytes in a valid data frame. If the number is incorrect, then the count is cleared and the message is discarded. If correct, the information is passed to the main program loop by setting a flag variable.

When the data line is first enabled, the distribution box has an internal delay of at least two byte times, which must elapse before any data will be sent. Any data received by the lamp will therefore be discarded as noise by the timer interrupt routine. After that, the real data can be safely sent down the line since the start bit of the first byte will be received correctly. When the transmission is completed, the line drivers will be disabled again.

Each of the cpus eg DCPU1, transfers data from the associated dual port RAM DP1 to the serial communication controller SCC1 to SCC30 with which it is associated one byte at a time, ie the first byte for SCC1 is transferred followed by the first byte for SCC2 and so on, each serial communication controller commencing transmission as soon as it has received its byte of data. The serial communication controllers operate to transmit data at 230.4 Kbps so that it takes about 35 μs to transmit each byte. Transfer of data from the dual port RAM DP1 to the serial communication controllers is, however, at a rate of several Mbps, so that the transmissions from all the serial communication controllers are almost simultaneous. The cpu DCPU1 is not required to monitor the transmission of data by the serial communication controller, but utilizes a software timer to commence transfer of the second byte to the serial communication controllers. This timer is started when transfer of the byte of data to the last serial communicationcontroller SCC30 has been completed and its time-out duration is slightly longer than the byte transmission time, say 40 μs. Transmission of all the messages takes about 1.5 ms out of a distribution unit main program loop duration of 4 ms.

As shown in FIG. 2, each lamp includes a serial communication controller 20 which controls reception of message data from the individual data link connecting it to the distribution unit 11. The receipt of any signal from the data link causes an interrupt of the lamp main cpu 21 (another MC68000) and the cpu 21 then controls acceptance of the signals. A timer 22 times the gaps between bytes received

from the data link and this timer causes another interrupt on time-out. The time-out time of the timer is between the times taken to transmit 1 and 2 bytes, so that time out always occurs following a spurious signal caused by line driver enabling. The time-out interrupt causes the cpu **21** to inspect the total number of bytes received since the initial interrupt and if this is less than the expected number of bytes (which is constant) the message is ignored. The time-out interrupt also resets a software data pointer to the beginning of a receive buffer in readiness for the next transmission.

The cpu **21** operates in accordance with programs stored in the lamp cpu ROM. On receipt of a message of valid length, a program variable representing the number of messages received since the lamp program was last started is incremented and the main program loop of the lamp cpu checks this variable every 16 mS. If the variable has changed since the last check, the data in the receive buffer is compared with corresponding values of variables representing current "desired values" of the various lamp function parameters. For example the receive buffer may contain two bytes representing the x, y and z co-ordinates of a point in an orthogonal three dimensional frame of reference, through which point it is required that the axis of the lamp beam should be directed. If the values of the corresponding byte pairs in the receive buffer and the desired value variables already contained in the cpu RAM are the same, no action is taken in respect of the control of the motors which control pan and tilt action of the lamp (to be described in more detail hereinafter).

As shown in FIG. 2, the main lamp cpu **21** communicates via serial data links **25***a*, **25***b*, **25***c* and **25***d* with four servo-control co-processors **26**, **27**, **28** and **29**. Each of these co-processors is a TMS77C82 cpu. Co-processors **26** and **27** respectively control pan and tilt operation, and each of the co-processors **28** and **29** can control up to six different dc servo-motors operating different functions of the lamp.

Before proceeding with a more detailed description of the circuitry and operation of the lamp electronics, some detail will be given of the various functions of the lamp. FIG. **8** shows the relative positions of a plurality of independently operable beam characteristic control elements within the lamp housing **100**. The lamp housing is pivotally mounted on a U-bracket **101**, which is itself pivotally mounted on a mounting base **102**. FIG. **9** shows the mounting base **102** which incorporates a pan drive motor/gearbox/optical encoder arrangement **104** which drives a gear **105** attached to the U-bracket via a reduction toothed belt drive **106**. FIG. **10** shows how, within the hollow structure of the U-bracket **101**, there is mounted a tilt drive motor/gearbox/optical encoder **107** which drives a gear **108** attached to the lamp housing via another reduction toothed belt drive **109**.

As shown in FIGS. **8** and **11**, within the lamp housing, a light source **110** is mounted within an ellipsoidal reflector **111** providing a light beam with an axis **112** which is reflected by a mirror **113**, which is a dichroic mirror that reflects only visible light and passes ultra voilet and infra red light, the reflected light passing out through an opening **114** at the opposite end of the housing. The reflector **111** has a generally cup-shape surrounding the bulb **1 10**. According to one aspect of the invention, the axis **112** has an angle pointing in a direction rearward relative to a perpendicular to the central axis **120** of the lamp unit. If the reflector is located as shown, such that an outside edge of the reflector is generally parallel to a rear end of the housing, the optimal packing efficiency is achieved. As shown in FIG. **8**, this allows the reflector to be most efficiently packed into the available space. The reflected beam from the mirror **113** passes firstly through a collimating lens **113***a*, and then the colour changer **115** which comprises dichroic filters having differing transmission characteristics mounted on co-centered three filter disks **115***a*, **115***b* and **115***c* rotable around a common axis of rotation. Each disk has nine different filters on it and one blank space around its periphery, so that up to 1000 different combinations of filters can be positioned across the beam by selective positioning of the three disks (although not all of these combinations are necessarily useful as some may block all visible light). The blank space of each of the disks can be used to eliminate any color changing characteristic of that disk. These disks are driven by three of the dc servo-motors. Next the light beam passes through the plane of a bladed shutter **116** (shown in FIG. **13**) and a first gobo wheel **117** which has various gobos mounted in or over circular holes therein. As shown in FIG. **12** described in more detail hereinafter, two motors are committed to driving the shutter **116** and the gobo wheel **117** respectively. Next, there is a second gobo wheel **118** on which there are mounted a plurality of gobos which are rotatable relative to the wheel **118**. There is one motor (not shown) for driving the gobo wheel **118** and another for rotating the gobos mounted thereon through a gear arrangement (not shown). Next along the light beam is a beam size controlling iris diaphragm **119** driven by another motor (not shown). Two further motors (not shown) drive two lens elements **120**, **121** along guides **122**, **123** parallel to the beam axis using lead screws **124**, **125**. The lens elements form a simple two element zoom lens controlling the spread and focus of the beam. Finally, an outer iris diaphragm **126** is provided adjacent the opening **114** and this is driven by a further motor (not shown). In the example described, therefore only eleven channels are actually employed.

Referring now to FIG. **12**, the shutter **116** is rotatably mounted on bearings **130**, **131** on a shaft **132** fixed to a mounting panel **133** which is secured to the housing. The gobo wheel **11 s7** is rotatably mounted on bearings on a tubular shaft **134** which acts to space the shutter **116** from a first drive gear **135**. The gobo wheel **117** is actually mounted on a second drive gear **136**. The shutter motor **137** (which is combined with a reduction gearbox and an optical encoder) is mounted on the panel **133** and drives a pinion **138** meshed with the first gear **136**. Similarly motor **139** drives a pinion **140** meshed with the second gear **136**. The shutter has four blades arranged symmetrically around its axis, with the blades and the gaps between them each subtending 45 degrees at the axis. The blades and the gaps between them are wide enough to block or clear the entire cross-section of the beam, shown in FIG. **13** at **116***a*.

Turning now to FIGS. **3** and **4**, the co-processor **26** is shown providing an eight bit data output to a d/a converter **40** (FIG. **3**) the output of which is amplified by an operational amplifier **41** and supplied to the "COMPEN" terminal of an LM3524 pulse width modulator ic **42** (FIG. **4**). The ic **42** control a P-channel enhancement mode MOSFET Q1 which, when switched on, connects a 24 V supply to a motor supply bus **43** through the intermediary of an inductor **44**. The motor is connected in a bridge formed by two push-pull pairs of MOSFETs Q2, Q3 and Q4, Q5. These four MOSFETs are driven by respective driver transistors Q6, Q7, Q8 and Q9. Transistors Q7 and Q9 are respectively controlled by "LEFT" and "RIGHT" outputs taken from the co-processor **26**, so that FETs Q2 and Q5 or FETs Q3 and Q4 are biassed to conduct. Transistors Q6 and Q8 are driven from a 40 V supply rail so as to ensure that FETs Q2 and Q4 are turned hard on when conductive, thereby ensuring minimum power dissipation in these devices.

The two FETs Q3 and Q4 are connected to the return bus via a current sensing resister RC, which supplies a current related signal to a voltage comparator **45** with hysteresis to provide an input to the **A6** input terminal of the co-processor **26** when the current exceeds a predetermined limit. This enables the co-processor to reduce the power applied to the motor to maintain it within safe operating limits.

The optical encoder of the pan motor provides two digital outputs in quadrature, these outputs being cleaned up by interface circuits and applied to two inputs of an HCTL-2016 counter ic 46 intended specifically for use with quadrature type encoders. The counter **46** counts up when the pulses are in one relative phase relationship and down when the opposite phase relationship exists. It therefore maintains a count-state related to the motor shaft position and hence the pan angle of the lamp. This count-state is applied to the C0 to C7 terminals of the co-processor **26**. The co-processor **26** also receives "desired value" data from the main lamp cpu **21**, via a 75176 ic 47 (which in fact serves both co-processors **26** and **27**). The ic 47 is used to control the transmission of data between the main lamp cpu and the co-processors. Normally the ic 47 is set to receive data from the cpu **21** and pass it to the two co-processors **26** and **27**. At power-up or when the main lamp cpu **21** transmits a "break" command, the co-processor **26** is reset by a circuit **48**. The co-processor **26** has a cycle time of 1 mS and on receipt of new data it determines the distance to be travelled and then increases the "desired position" value which is compared with the actual position count by one sixteenth of the required change on each successive iteration of its control loop.

The desired value signals passed from the cpu **21** to the co-processor **26** are also time-sliced, being incremented every 16 mS. When new position data is transmitted to the lamp it is accompanied by data representing the length of time over which the movement is to be spread. The data is received, as mentioned above, in the form of two byte numbers respectively representing the x, y and z co-ordinates of a point in a Cartesian co-ordinate system. During initial setting up of the system, each lamp is sent data which informs its cpu **21** of its position in the coordinate system and also of its orientation.

On receipt of a new set of "point at" co-ordinates, the cpu **21** undertakes a "time-slicing" operation to determine how data should be passed to the co-processors **26** and **27**. First of all, it determines how many 16 mS loops will take place in the time duration determined by the data contained in the massage received by the lamp and sets up a variable U equal to the reciprocal of this number. A travel variable P is initialised to zero and the total distance to be travelled is determined for each of the pan and tilt movements. Thereafter, on every iteration of the 16 mS loop the travel variable P is incremented by the reciprocal variable U, the result is multiplied by the total travel required and this is added to (or subtracted from) the previous desired value before transmission to the co-processor **26** or **27**. When the variable P exceeds unity, the target has been reached.

The message sent to the lamp may include a flag indicating whether travel is to occur in a linear fashion as described above or have a sinusoidal profile imposed on it. In the latter case the value of P is modified as follows:

$$P'=\sin(2*P)+0.5*(P>0.5)$$

the latter term being 0 or 1

The main cpu **26** must next convert the x,y,z values into pan and tilt value data for passing to the co-processors **26** and **27**. The cpu first carries out a linear transformation of the absolute x,y,z co-ordinates into co-ordinates x',y',z' relative to the lamp's own frame of reference using the data supplied during initial set up. The ratio of the transformed x' and y' values is calculated as a 16-bit integer, which is used as an index to an ARCTAN table stored in ROM to obtain a value for the desired pan angle. To find the tilt angle, it is first necessary to establish the radial position of the target point in the transformed horizontal plane by calculating the square root of the sum of the squares of the co-ordinates x' and y'. In carrying out this calculation it is necessary to detect an overflow condition which exists if the sum of the squares is a 33 bit number. If this condition is detected, each square is divided by four and a new sum is formed, an overflow flag being set to indicate that overflow has occurred. The square root is found by up to sixteen steps of successive approximation and the result is doubled if the overflow flag was set during the calculation. The resulting square root is divided by the value z' and the result is applied as before to the ARCTAN table to determine the tilt angle. The results obtained represent the new pan and tilt positions to which the lamp is to be moved.

The arrangement described for sending out x, y and z co-ordinate data instead of pan and tilt angle data is highly advantageous in that it enables the console main cpu load to be significantly reduced and also makes it very easy for a console operator to control light beam movements. It is frequently required for a group of lamps to be used together to illuminate a single performer. Where the performer moves from one position on stage to another it is required for all the lamps to change position simultaneously to follow. If the system involved transmission of pan and tilt angle data, this data would be different for every lamp in the group. It would have to be set up by the console operator and stored in cue files on the hard disk drive unit **15**. This would be a very time consuming operation as the pan and tilt angles for each lamp would have to be established and recorded individually. The cue record would need to be of considerable size to record all the different data for each lamp. With the arrangement described above, however, only the x,y,z co-ordinate data needs to be stored and when the cue is recalled the same data is sent to each of the lamps in the group.

Whilst it is theoretically possible to use stored cue data in x,y,z co-ordinate form and to use the console main cpu **14** to calculate the pan and tilt angles to send to the lamps, this would be unsatisfactory as the calculations involved would impose a very heavy load on the cpu **14**, particularly where a large number of lamps in several different groups had to be moved as the result of a single cue.

As described above a "point-at" mode is envisaged as the normal operating mode. However, other modes of operation are also envisaged. For example, the lamp could be instructed to point away from the point specified or to point in a direction parallel to a line joining a fixed point (eg the origin of the co-ordinate system) to the point specified. These "point-away" and "point parallel" modes would be selected by means of flags included in the data transmitted to the lamps.

The arrangement described enables the lamps to be very precisely synchronised. The data is transmitted from the distribution unit to all of the lamps simultaneously and each lamp can start to respond at the end of the message. This enables very precise direction of all the lamps to a moving point in "point-at" mode and very clean parallel sweeps to be made in "point parallel" mode.

It should be noted that the use of x,y,z co-ordinates is also very advantageous in situations where a pre-arranged lighting performance is to be used in several different venues.

The pre-loaded gantries or trusses used for such touring performances cannot always be mounted at exactly the required positions relative to the stage because of local conditions. In this case all that is needed is for offsets data to be sent to the lamps at set-up time to enable each lamp cpu to correct its position data. No editing of the individual prerecorded cues is necessary as it would be in the same circumstances if pan and tilt data were stored.

As part of the set-up procedure for each performance it is necessary to initialise the values of the actual pan and tilt angle count-states, since encoders of the type used do not give any absolute position data. This is accomplished by driving the lamp to an end stop in one direction for each movement. The lamp is driven back to a predetermined number of counts and the counters are reset to zero at this position.

Turning now to FIGS. 5 to 7, the circuitry for controlling the individual dc servo-motors inside the lamp is more complex as each co-processor has to deal with up to six servo-motors. As shown in FIG. 5, the co-processor **28** controls a number of data routers **50** to **54** which determine which channel is being controlled at any given time. The router **50** co-operates with six HCTL-2016 counters **55** which count the quadrature pulse outputs of the respective encoders, to determine which of the counters should supply its count-state to the co-processor **28**. Router **51** controls individual resetting of the counters **55**. Router **52** co-operates with a 74HC175 ic **56** (one for each channel) to determine which L6202 ic motor controller **57** is enabled and also routes "RIGHT" and "LEFT" signals from the co-processor to the circuits **57**. Router **53** controls routing of position error data calculated by the co-processor **28** for each channel to latches **58** (one for each channel) at the input of pulse width modulator circuits for controlling the motor controllers **57**. This error data is actually passed to the latch **58** in an inverted form, so that the larger the error, the smaller the value passed is. Router **54** routes various digital sensor signals to a sensor input of the co-processor, Such sensors are utilized by some of the channels to indicate when the moving part in question is in a datum position. This is required for the gobo wheels, the colour wheels and the shutter, but not for the iris diaphragms or lenses which can be moved to end stop positions. During datum set-up the sensors (optical sensors sensing a hole or flag or Hall effect sensors) are detected and the HCTL counters are reset.

As co-processor **28** has only 256 bytes of internal memory, extra memory is required for each channel to store program variables. The RAM selection control circuit is shown in FIG. 7. The memory ic **59** (an HM6116LP ic) has 11 address lines of which eight are connected to the co-processor write bus via a latch circuit **60** and the remaining three of which are connected to spoare outputs of three of the ics **56**. Spare outputs of the selectors **50, 51, 52** are connected to control terminals of the memory ic and a spare output of the selector **53** is connected to an output enable terminal of the latch circuit **59**. Thus a particular address in the memory ic can be selected by the co-processor by first setting the ics **56** and the selectors **50, 51, 52** to appropriate states and then outputting the lower bytes of the address to latch **60** whilst output from latch **60** is enabled. Two further eight bit latches **61** and **62** provide temporary storage for data to be written to and data just read from the memory ic **59**. When neither reads nor writes are required the memory data bus is tri-stated. Bus contention is thus avoided.

Circuit **57** actually controls the motor current, but it in turn is controlled by a pulse width modulator circuit, comprising the latch **58** and a digital comparator **65** which compares the contents of latch **58** with the count-state of an 8-bit continuously running counter **66***a*, **66***b* serving all channels. The comparator output goes high when the count-state exceeds the latch contents, so that if the latch content is low the comparator output is high for a high proportion of each cycle of the counter **66***a*, **66***b*. The output of the comparator **65** is ANDed with an enable output from ic **56** by a gate **67** and then with the output of an overcurrent detector circuit **68** by another gate **69**.

When a new target value for one of the parameters controlled by co-processor **58** arrives in the receive buffer, and it is associated with execution duration data (this may apply to lens movements, colour changer movements, gobo movements and iris diaphragm movements, but not shutter movements) the cpu **21** handles time slicing as in the pan and tilt operations. Since several channels are controlled by each co-processor, however, no interpolation by the co-processor is used. Instead each channel has its error checked and a new value written (if necessary) to latch **58** every 12 mS

In the case of the shutter, the message received by the lamp merely includes a shutter open or shutter closed command. When the required shutter status changes, the main cpu merely increases the target shutter angle by 45 degrees (in the case of a four bladed shutter) and passes the new value to the co-processor.

This arrangement enables the shutters of some or all of the lamps to be operated in synchronism. Moreover, the console cpu **14**, can operate to update the shutter open/closed instructions at regular intervals to obtain a stroboscopic effect, synchronised for all the lights.

What is claimed is:

1. A coordinate converting lamp unit, comprising:

a communication element, receiving a signal for the lamp unit, said signal including information indicative of a desired position of the lamp in an absolute coordinate system; and

a first lamp, receiving said signal including said desired position information, and including:

a) a calculating device, including information from which a conversion between said absolute coordinate system and a first coordinate system of the lamp can be effected, and operating to convert said desired position information from said absolute coordinate system into said first coordinate system of the first lamp; and

b) lamp moving devices, said lamp moving devices responsive to said information in said lamp coordinate system to move said lamp to a position indicated thereby and to thereby point said lamp at an area indicated by said absolute coordinate system.

2. A lamp unit as in claim **1**, further comprising:

a second lamp, receiving the same said signal including said desired position information in said absolute coordinate system, and including:

a) a calculating device, including information from which a conversion between said absolute coordinate system and a second coordinate system of the second lamp can be effected, said second coordinate system of the second lamp being different than the first coordinate system of the first lamp, and operating to convert said desired position information into said second coordinate system; and

b) lamp moving devices, said lamp moving devices responsive to said information in said second lamp coordinate system to move said lamp to a position indicated thereby to thereby also point said second lamp at said area indicated by said absolute coordinate system.

3. A distributed lighting system comprising:

a lighting console, storing a plurality of control information for a plurality of remotely-located luminaires, said lighting console including a memory area storing control information for a plurality of said luminaires;

a plurality of serial communication controllers, said serial communication controllers producing serial outputs, a separate serial output being provided for each of said luminaires, and each said serial communication controller associated with a block of said memory, each serial communication controller including a line driver which is enabled to transmit data to its associated luminaire; and

a plurality of remotely-located luminaires, each said luminaire connected to a different one of said serial outputs, each luminaire comprising a plurality of motors respectively controlling pan and tilt of the luminaire, each said luminaire including a serial communication controller, connected to receive information from an associated serial output and to produce an output indicative of the information on said serial output, said output including information which indicates coordinates in space of a location through which an axis of a light beam from the luminaire will be directed, said luminaire including a processor which commands said motors to pan and tilt the luminaire to direct said axis of the beam to the coordinates in space.

4. A system as in claim 3 further comprising a plurality of coprocessing devices, each said coprocessing device associated with a different motor control function of the luminaire.

5. A system as in claim 4 wherein said motors include a panning motor and a tilting motor, and wherein said coprocessing devices include a first coprocessing device controlling said panning motor and a second coprocessing device controlling said tilting motor.

6. A system as in claim 3 further comprising a plurality of coprocessors, each coprocessor associated with a motor and controlling operations of said motor.

7. A system as in claim 3 further comprising a detector which detects a turn on of said line driver and determines synchronization of the frame being sent thereby.

8. A computer controlled automated moving luminaire operating responsive to commands from a central console, comprising:

a lighting system;

a first motor system which controls a direction of said lighting system to move in a panning direction;

a second motor system controlling said lighting system to move in a tilting direction;

a communication controller, receiving communication information from the central console;

a processor, receiving said information from said communication controller and translating said information into commands for pan and tilt;

a first coprocessor, associated with panning carried out by said first motor system, said first coprocessor receiving said information indicative of panning, and controlling said first motor system to appropriately pan the lighting system; and

a second coprocessor, associated with tilting carried out by said second motor system, and controlling said second motor system to appropriately tilt the lamp.

9. A method of synchronizing positions of a plurality of moving lights, comprising:

issuing a control command from a central console to a plurality of remotely-located moving lights, said control providing position information to said moving lights and also providing at least one time value indicating a time when said moving light should be located at a position;

receiving said control command in each of said plurality of remotely-controlled moving lights;

each of said remotely-controlled moving lights determining, a movement profile from said position information and time which will cause each of said remotely-controlled luminaires to point at substantially a same position at the same time.

10. A method as in claim 9 wherein each said moving light includes a processor and each said processor carries out a time slicing operation to provide said movement profile.

11. A method as in claim 9 further comprising, in each moving light, obtaining said position information, and translating said position information into a local set of coordinates in the lamp's own frame of reference, and calculating positions for moving the lamp based on said local set of coordinates.

* * * * *

EXHIBIT E

US005969485A

United States Patent [19]

Hunt

[11] **Patent Number: 5,969,485**

[45] **Date of Patent: Oct. 19, 1999**

[54] **USER INTERFACE FOR A LIGHTING SYSTEM THAT ALLOWS GEOMETRIC AND COLOR SETS TO BE SIMPLY RECONFIGURED**

[75] Inventor: **Mark A. Hunt**, Birmingham, United Kingdom

[73] Assignee: **Light & Sound Design, Ltd.**, Birmingham, United Kingdom

[21] Appl. No.: **08/753,035**

[22] Filed: **Nov. 19, 1996**

[51] **Int. Cl.**^{6} **H05B 37/02**

[52] **U.S. Cl.** **315/292**; 315/316; 315/294; 315/297; 315/324

[58] **Field of Search** 315/316, 314, 315/294, 297, 292, 324

[56] **References Cited**

U.S. PATENT DOCUMENTS

3,706,914	12/1972	Van Buren	315/316
3,766,431	10/1973	Isaacs	315/316
3,784,875	1/1974	Baker et al.	315/316
4,358,715	11/1982	Dinges et al.	315/316
5,010,459	4/1991	Taylor et al.	315/316
5,530,322	6/1996	Ference et al.	315/316
5,621,282	4/1997	Haskell	315/316
5,668,537	9/1997	Chansky et al.	315/316

OTHER PUBLICATIONS

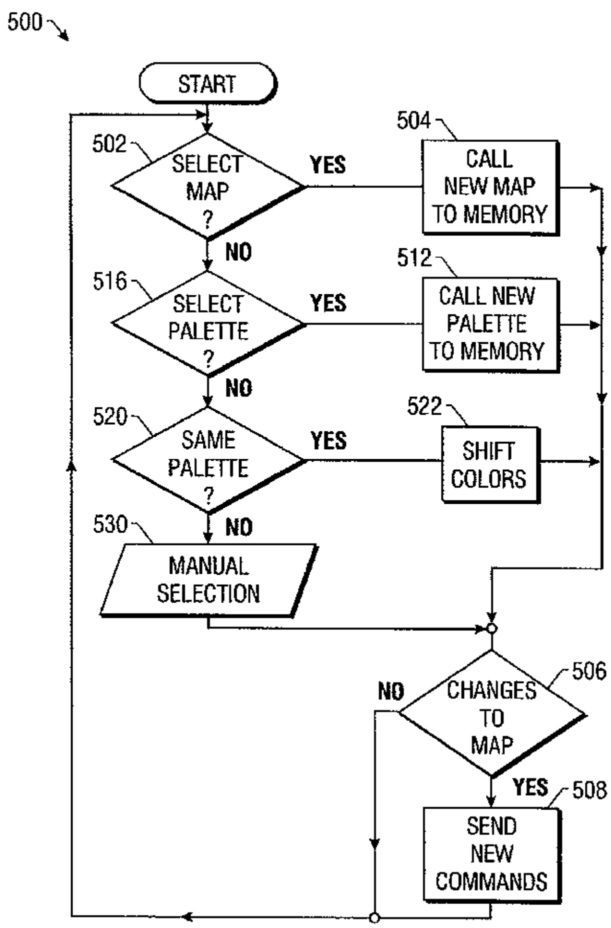
Epperson "A Digital Lighting System for the Theatre". Theasn Yale School of Drama 1974.

Primary Examiner—Michael B Shingleton
Attorney, Agent, or Firm—Fish & Richardson P.C.

[57] **ABSTRACT**

Lamps forming a lighting show are grouped into maps. Each map includes an association between the lamps and specific sets. Lamps are in particular sets in particular maps and can be in different sets in other maps. Each set can be associated with a parameter for that set. One such parameter is the color for the set of lamps. Both maps and parameters can be changed by a single key press. This allows single key press parameter cycling.

17 Claims, 4 Drawing Sheets



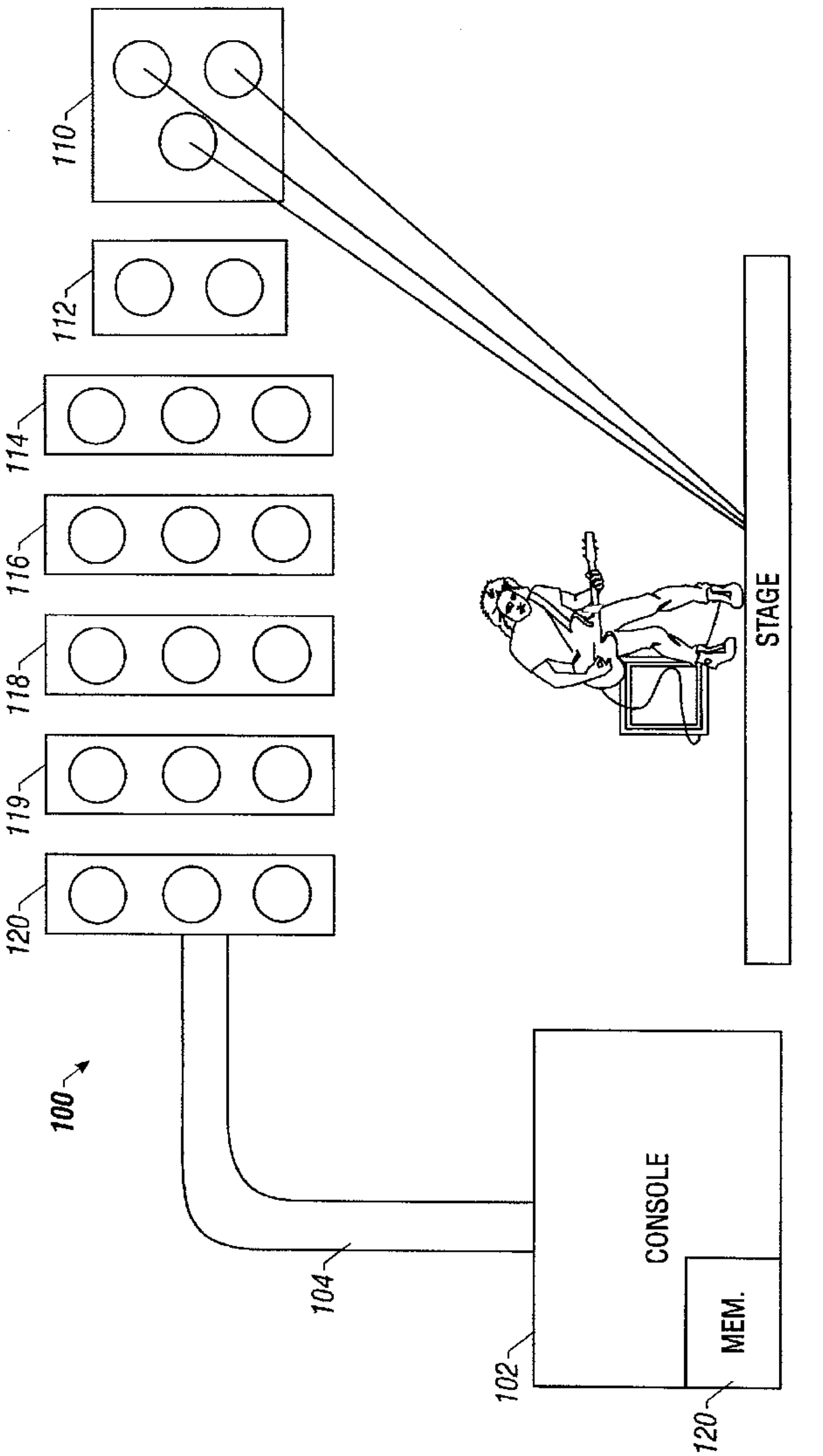


FIG. 1

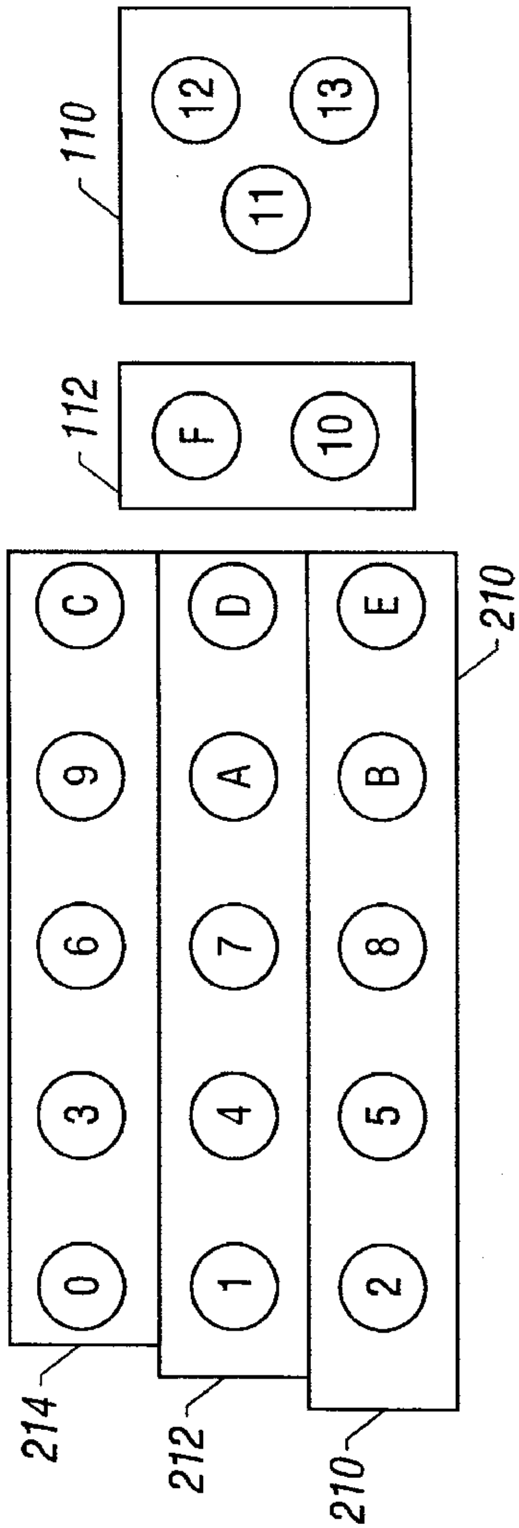


FIG. 2

	(SET)	(LAMPS)
	1	0, 1, 2
119	2	3, 4, 5
118	3	6, 7, 8
116	4	9, A, B
114	5	C, D, E
112	6	F, 10
110	7	11, 12

Fig 3A

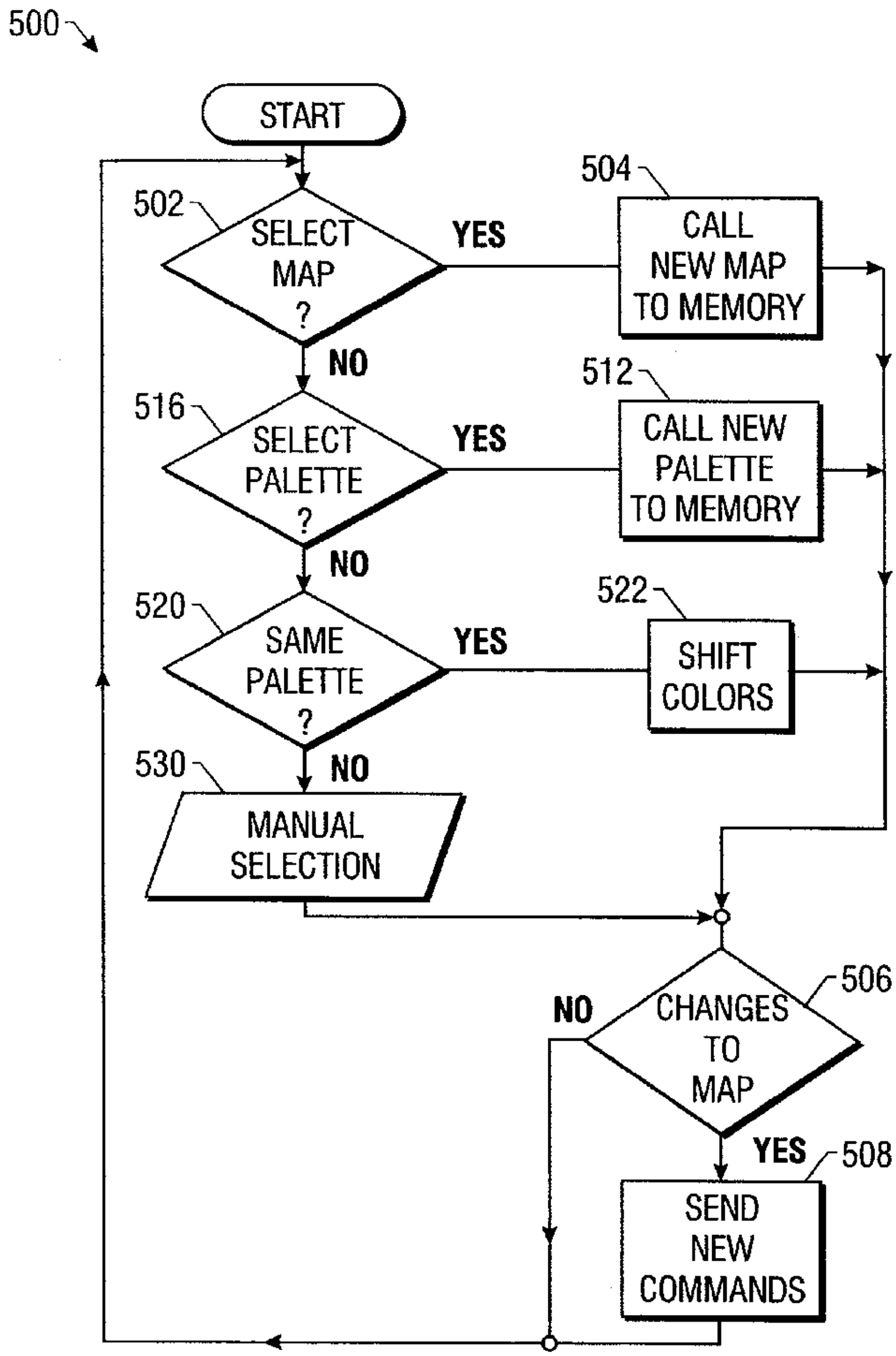
	(SET)	(LAMPS)
210	1	2, 5, 8, B, E
212	2	1, 4, 7, A, D
214	3	0, 3, 6, 9, C
110	4	F, 10
112	5	11, 12, 13

Fig 3B

PALETTE	COLOR	SET
1	00	1
	0A	2
	06	3
	1F	4

PALETTE	COLOR	SET
2	01	1
	0B	2
	0F	3
	3F	4

Fig 4



500
START
502
SELECT MAP ?
YES
504
CALL NEW MAP TO MEMORY
NO
516
SELECT PALETTE ?
YES
512
CALL NEW PALETTE TO MEMORY
NO
520
SAME PALETTE ?
YES
522
SHIFT COLORS
530
NO
MANUAL SELECTION
NO
CHANGES TO MAP
506
YES
508
SEND NEW COMMANDS

FIG 5

USER INTERFACE FOR A LIGHTING SYSTEM THAT ALLOWS GEOMETRIC AND COLOR SETS TO BE SIMPLY RECONFIGURED

FIELD OF THE INVENTION

The present invention relates to a combination driver for grouping stage lighting parameters into sets within maps and allowing changing the combinations formed by the contents of the parameters and the maps. More specifically, the preferred embodiment describes a plurality of luminaires which are dynamically arranged in maps, each map assigning the luminaires to specific sets, and the parameters being color palettes assigning colors to the sets.

BACKGROUND AND SUMMARY

Stage lighting is increasingly becoming an important part of theatrical productions, such as rock and roll concerts or theater presentations. A modern stage lighting effect uses a computer to choreograph the lighting effects to be initiated and carried out at pre-planned times. The choreographed effect has usually been planned in advance.

The choreographed effect is usually planned between the artist, often the lighting designer, and the console operator. A dry run through the show is conducted while the lighting designer decides what lighting effects are desired at different parts of the show.

The console operator controls the lighting system according to the lighting designer's direction, and by so doing plans the lighting effects that occur at different times during the show. Those lighting effects need to be carried out by the console operator.

The lighting designer will often want to try different effects to see what they look like and how they will fit in. Each attempted effect requires the console operator to arrange the operation of each light in the way that the lighting designer has requested.

For example, the lighting designer may have in mind a certain effect to be carried out in primary colors. If there is a desire to see what certain parts of that effect would look like in pastel colors, the console operator will need to change a number of different sets of lights to pastel colors. The console operator needs to do this as quickly as possible, but each light may need to be seperately controlled.

The present invention recognizes this problem, and devises a system which enables simple button presses on the console to command combinations of effects to facilitate the console operator's chore during this operation. One such feature allows cycling through many different kinds of lighting effects and grouping effects.

Present technology has necessitated that most, if not all, lighting shows be conducted automatically, based on information that has been stored in advance. This has made it difficult to improvise the lighting effect during a lighting show.

According to one aspect of the present invention, a number of stage lights form a show. The stage lights are defined into at least two different maps. Each map includes a set assignment for each of the stage lights. Each map includes at least some of the different stage lights in different sets.

A parameter palette is formed which includes parameters for the different sets. A preferred parameter palette is a color palette. For example, a primary color palette could change all of the lights in set **1** to red, the lights in set **2** to green, and the lights in set **3** to blue. A pastel palette, on the other hand, changes the lights in set **1** to pastel pink, the lights in set **2** to pastel blue, and the lights in set **3** to pastel green. Other different palette sets are also possible. Hence these different sets have different colors associated with the lamps in the set, thereby allowing different combinations of lamps to colors.

When a specific palette is chosen, each color in the palette can be applied to a set in the current map. A particularly preferred technique allows each palette and each map to be changed by a single key press.

Another part of this technique rotates the combinations, i.e., it rotates the different sets through the colors within the palettes. Therefore, the different palettes, which include parameters of predetermined types, can be rotated through the different sets either at random or in an organized fashion to allow the different parameters to be assigned to different sets and to test that effect. The maps, i.e., the associations between the lamps and the groups, can be rotated in a similar way.

Another aspect groups the lamps forming a lighting show into maps. Each map includes an association between the lamps and specific sets. Lamps are in particular sets in particular maps, and can be in different sets in other maps. Each set can be associated with a parameter for that set. One such parameter is the color for the set of lamps. Both maps and parameters can be changed by an association changing device, e.g., a single key press. This allows single key press parameter cycling.

All of these operations are automatically carried out using simple keystrokes to form the different combinations. This hence allows the associations to be carried out in a shorter time. No revenue is derived from this rehearsal time, hence increasing the economic incentive for shortening this time.

BRIEF DESCRIPTION OF THE DRAWINGS

These and other aspects of the present invention will be described with reference to the accompanying drawings in which:

FIG. **1** shows a basic block diagram including a number of lights, their relationship with the stage, and their relationship with a console;

FIG. **2** shows an alternative map which groups the lamps into different sets;

FIGS. **3**A and **3**B show the stored memory information for these maps;

FIG. **4** shows a memory map of color pallette information; and

FIG. **5** shows a flowchart of operation of the combination forming technique of the present invention.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

The basic system of the present invention is a stage lighting system as shown in FIG. 1. The FIG. **1** stage lighting system includes a lighting rig, which includes a group of luminaires ("lamps") forming a lighting show. The lighting rig **100** shown in FIG. **1** includes 20 lamps. In actual practice, a lighting show would actually include more than 20 lamps, but 20 is sufficient to illustrate the concept. The lighting rig **100** can be considered as a set of potential patterns.

Lighting rig **100** is controlled by console **102** through lighting cable **104**. The preferred console is an ICON

CONSOLE™ made by Light & Sound Design to control lighting rig **100** using Light & Sound Design's proprietary ICON™ format. However, any other console, including consoles available from other lighting companies could be used for this purpose.

The ICON CONSOLE™ **102** is a computer-based system which operates according to a stored program. The microcomputer used in this system is an M68000 which produces outputs according to the stored program. The outputs produces a control for each of the luminaires in the group. Each lamp receives commands to control its movement, position, color, specific light pattern to be projected (gobo), focus, dimmer, and iris. Any of these parameters, and any other parameters that are controlled by a lamp, could be controlled by forming the combinations described according to the present invention. The preferred embodiment described herein chooses the color parameter. However, it should be understood that any of these parameters could be controlled.

The entire lighting rig **100** is then arranged into predetermined maps. Each map is formed of a plurality of sets, and each set includes a number of different lamps—a pattern of lamps. Some of the sets may be formed of groups of lamps that are always used together, for example, set **110** may be a group of three lamps which shines on the same spot and hence would normally be used altogether. Other sets may be dynamically changed. Each map is essentially a view of the geometry of the lighting rig, with each subgeometry within the map being a set.

Each map is a group of sets. Each set is an association between the lamps and their set association. FIG. **1** shows the lighting rig **100** arranged into a first map. This map includes groups **110**–**120**. Each of the lamps within the rig is assigned to a specific set.

FIG. **2** shows another map, which we will call map **2**. Map **2** includes different set associations than map **1**; some of the sets are the same as map **1** and others are different than map **1**. Some of the lamps may be within the same set, such as sets **110** and **112** which are the same in map **1** and map **2**. The other sets **210**, **212**, and **214** include different groupings for the lamps. Note that the different sets in map **2** form a different geometrical pattern than the geometrical patterns in set **1**.

Of course, in actual practice there would be more than two maps.

The maps are assigned in advance and stored within the console memory, e.g., as computer data. Each lamp has a pre-assigned serial number. In this embodiment, the serial numbers, for simplicity, are designated as 0_H through 13_H. The number of sets in this embodiment might be limited to $16(F_H)$, although there is no practical limit on the number of sets which could be assigned.

Memory **120** within console **102** stores a relationship between each set number 0_H through F_H and the lamps within that set. For instance, the memory map for map **1** is shown in FIG. **3**(*a*) and the memory map for map **2** is shown in FIG. **3**(*b*).

A number of predefined palettes are also used according to the present invention. Each palette has multiple values defining a whole set of parameters, here a whole set of colors. Each parameter, for example, has a number of different forms.

According to the present invention, 256 different colors are defined. Each of the colors is assigned with a number. Each number represents a specific color that is available from the ICON™ lamp. The colors may range between 0 and FF_H.

The palettes are groups of colors which are in some way related to one another. Exemplary palettes include primary color palettes, such as red, green, blue; pastel color palettes; highly saturated palettes; weakly saturated palettes; rainbow palettes of colors that form a rainbow; random color palettes, single-color palettes, such as differing hues of red, differing hues of blue; dual-and triple-color palettes; and any other palette of lights that might go together. Each palette can include up to 16 colors.

The basic color palette that is stored in memory is shown in FIG. **4**. The memory map includes information for the different numbered palettes. Note that each color in the FIG. **4** color palette is associated with a set number.

The operation of operating the lights is shown in the flowchart of FIG. **5**. Step **500** starts the process with a determination at step **502** whether a selection of map has been requested. If so, the new map is called into memory at step **504**. The map stored in memory is of the form shown in FIGS. 3A and 3B—the table includes the serial number of each lamp, and its set association for various parameters of that lamp, including, but not limited to, color, focus, position, and the like. Control then passes to the map change operation steps. Step **506** determines if there are any changes to the memory. If changes are detected, appropriate messages are sent at step **508** commanding the lamps to their new color. Flow then returns to the main loop.

Step **510** determines a selection of a palette. If there is a selection of a new palette at step **510**, the palette is called to working memory at step **512**. Control then passes to the change detection routine, which processes the changes according to steps **506** and **508**.

This results in a lamp-table state in which the default combinations of the selected palette as shown in FIG. **4** has been associated with the sets within the selected maps. Now the colors can be changed in a number of different ways. Step **520** represents selecting the same palette again. Reselection of this same palette causes the same palette to be used, but the colors to set combinations to shift. This can be carried out in a number of different ways according to the present invention. The most preferred way is by Fourier-bit swapping. Each of the colors is associated with a set and a Fourier technique is used to rearrange the bits within the set so that each color is in a definable, yet pseudo-random way, associated with a different set.

In the event that there are less colors than there are a number of sets, the colors can be simply re-used for the new sets. In the opposite scenario, where there are fewer sets than colors, certain of the colors within the palette will be unused at different times.

Another alternative for cycling is a hash algorithm.

Another technique, less preferred but also useable, is to use a pseudo-random number generator to select numbers between 0 and F_H. Yet another technique simply shifts the relationship between the colors and the sets in an ordered fashion so that the color previously associated with set **1** is now associated with set **2**, and the color previously associated with set f becomes with set **0**.

Yet another technique uses a factorial association technique. N colors have N! different available combinations. The N! combinations are associated with the different sets.

At step **522**, the shifted colors are defined into working memory, followed by the change processing routine of steps **506**/**508**.

Step **530** enables manual selection of certain groups/ colors. This selection allows certain sets to be manually

selected. Those manually-selected groups are maintained at the manually-selected color until cancelled. The other group combinations can be shifted using the same technique previously described.

Any desired effect includes a number of elements within memory stored as a table. That table can then be stored as a cue for the desired effect.

Although only a few embodiments have been described in detail above, those having ordinary skill in the art will certainly understand that many modifications are possible in the preferred embodiment without departing from the teachings thereof.

All such modifications are intended to be encompassed within the following claims.

What is claimed is:

1. A stage lighting system, comprising:

a plurality of electrically-controllable lights, each including a control path over which said lights can be commanded from a remote location; and

a controller, connected to said lights and electrically controlling said lights according to a stored program and user interface, said controller including:

a memory, storing a plurality of maps for said lights, each of said maps assigning said lights to light sets and at least a plurality of said light sets including more than one light, and a plurality of parameter sets of light-controlling parameters for said light sets including a pallette of colors for said light set,

an association between light set parameters from said parameter sets which are associated with said light sets, said controller commanding said lights in each said light set based on the parameters associated with the light sets, and said user interface including an association changing device, which has a first keypress which selects and changes at least one of said maps in said memory which is being used, said map being selected to select all of the groupings of lights to light sets in that map at one time said first key press cycling through maps with a single key press causing a single cycling, and an association between said sets and colors in said pallette which is selected according to a single key press all one to cycle through an association between light sets and parameters.

2. A system as in claim 1, wherein said association changing device is a button, which when pressed, changes said at least one of said maps and said sets.

3. A variable lighting system, comprising:

a plurality of lights;

a memory, storing at least two different maps, each said map assigning said lights to sets, each map includes at least some of the different lights in different sets, and said memory including a plurality of parameters associated with said sets including at least first and second different color palletes each of which is group of colors used together; and

a controller, having a first control element allowing selection of one of said maps controlling said lights such that each said set is controlled by the parameters associated with said each set for all of one map when selected, and a second control element which controls cycling between said first and second color pallettes, a first association between said map and said first color pallette, and a single key press changing said association to between said map and said second color palette.

4. A system as in claim 1, wherein a first color palette includes primary colors, and a second color palette includes colors other than primary colors.

5. A system as in claim 1, wherein said association changing device is a key, and each step in the rotation is commanded by a single key press.

6. A system as in claim 3, wherein said parameters include at least movement, position, color, specific light pattern to be projected (gobo), focus, dimmer, and iris.

7. A system as in claim 3, wherein said map stores a numerical value associated with each parameter.

8. A system as in claim 7, wherein said map stores an association between a numerical value indicating a specific light, and the numerical value associated with each parameter.

9. A system as in claim 3, wherein said parameter set is a palette with specific colors; each color being associated with one of said sets from a selected map.

10. A variable lighting system, comprising:

a plurality of lights;

a memory, storing a first geometrical arrangement of said lights, and a second geometrical arrangement of said lights different than said first geometrical arrangement of said lights, each geometrical arrangement of lights including subgeometries therein, and said memory storing an association between at least one subgeometry among said subgeometries and a parameter for said one subgeometry; and

a changing element, allowing changing a parameter which is associated with all lights in said subgeometry using a single key press.

11. A system as in claim 10, further comprising

a controller, controlling said lights such that each said subgeometry is controlled by the parameter associated with said each subgeometry.

12. A method of controlling a variable lighting system, comprising:

assigning each of a plurality of lights to respective sets;

associating one of a plurality of parameters with each said set;

using a parameter key press to rotate the association between the parameters and the sets; and

controlling said lights such that each said set is controlled by the parameter associated with said each set.

13. A system as in claim 3, wherein said parameter set is a palette with specific colors; each color being associated with one of said sets from a selected map.

14. A method as in claim 12, further comprising changing a set of parameters which forms said plurality of parameters.

15. A method as in claim 12, wherein said plurality of parameters is a color palette.

16. A method as in claim 12, wherein said assigning comprises storing an association between a numerical value indicating a specific light, and a numerical value associated with each parameter.

17. A method of controlling a variable lighting system, comprising:

assigning each of a plurality of lights to respective sets;

associating light controlling parameters with said sets;

changing an assignment between lights and sets to change the light controlling parameter associating between said sets and said parameters; and

controlling said lights such that each said set is controlled by the parameter associated with said each set.

* * * * *

EXHIBIT F

US006011662A

United States Patent [19]

[11] Patent Number: 6,011,662

[45] Date of Patent: Jan. 4, 2000

Evans

[54] **CUSTOM COLOR WHEEL**

[75] Inventor: **Nigel Evans**, West Midlands, United Kingdom

[73] Assignee: **Light & Sound Design, Ltd.**, Birmingham, United Kingdom

[21] Appl. No.: **09/108,777**

[22] Filed: **Jul. 1, 1998**

[51] **Int. Cl.**[7] **G02B 5/22**

[52] **U.S. Cl.** **359/891**; 359/892; 359/885; 362/277; 362/282; 362/283

[58] **Field of Search** 359/885, 891, 359/892; 362/277, 280, 282, 283, 322, 323, 455

[56] **References Cited**

U.S. PATENT DOCUMENTS

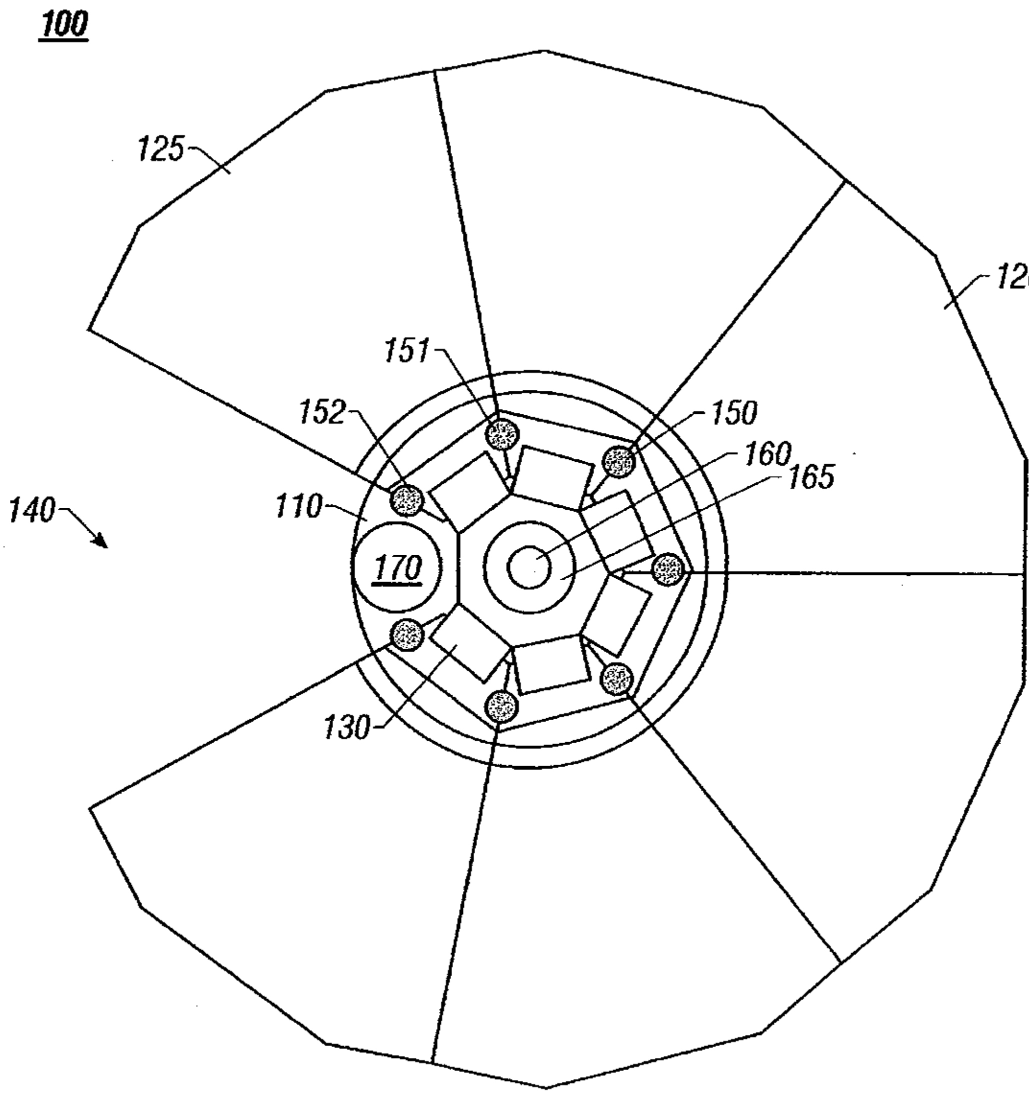
5,060,126 10/1991 Tyler et al. 362/277

Primary Examiner—Cassandra Spyrou
Assistant Examiner—Craig Curtis
Attorney, Agent, or Firm—Fish & Richardson P.C.

[57] **ABSTRACT**

A color wheel for lighting fixtures comprising a hub wherein removable, user-selected dichroic filters are engaged by a spring element. The filters are removably held by two pegs extending upward and one nub extending downward. The filters can be lifted for removal.

20 Claims, 7 Drawing Sheets



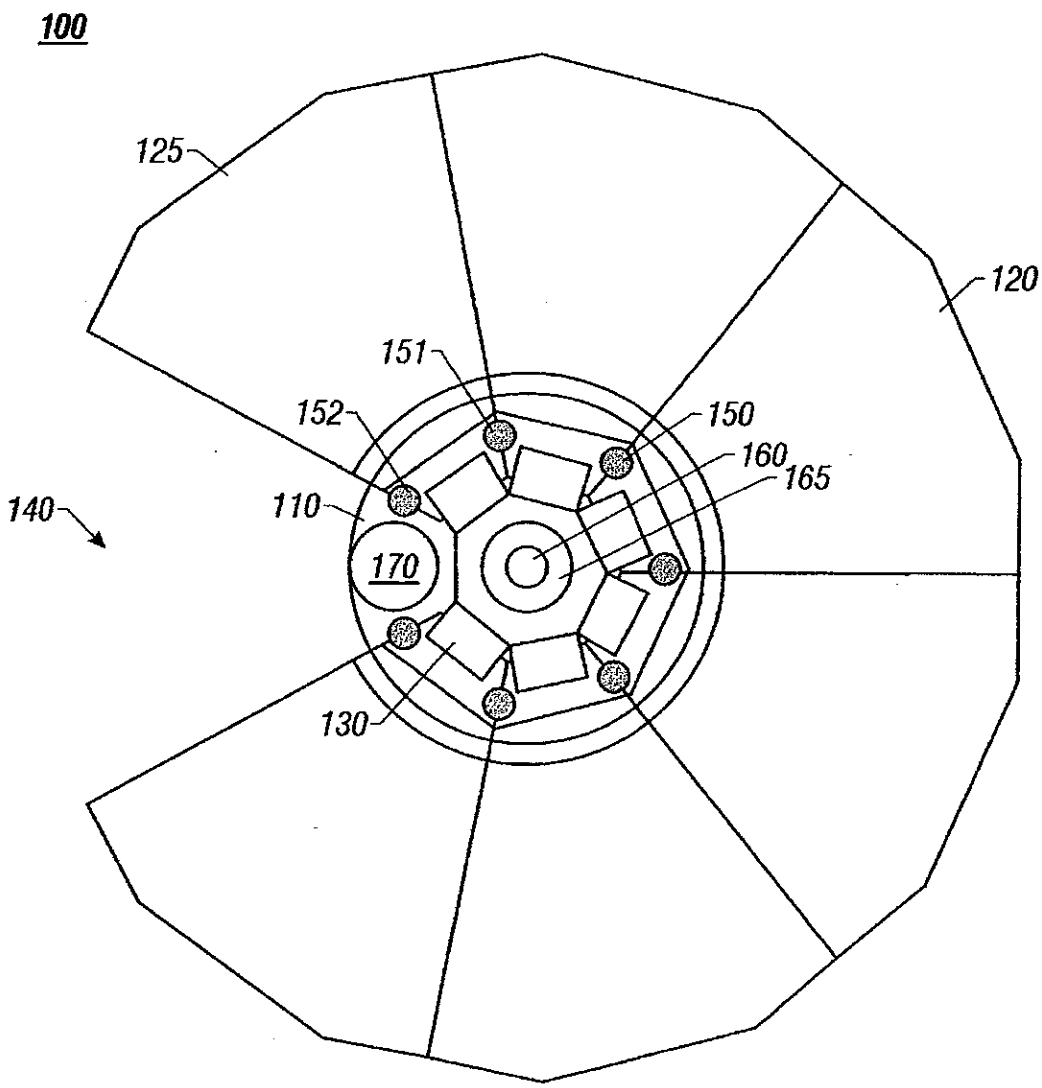
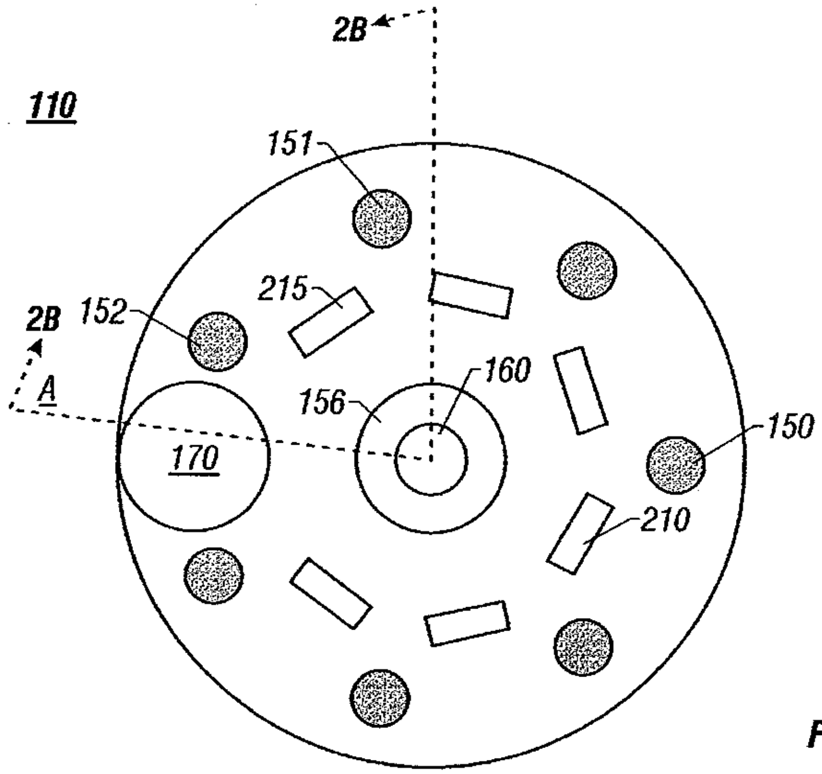
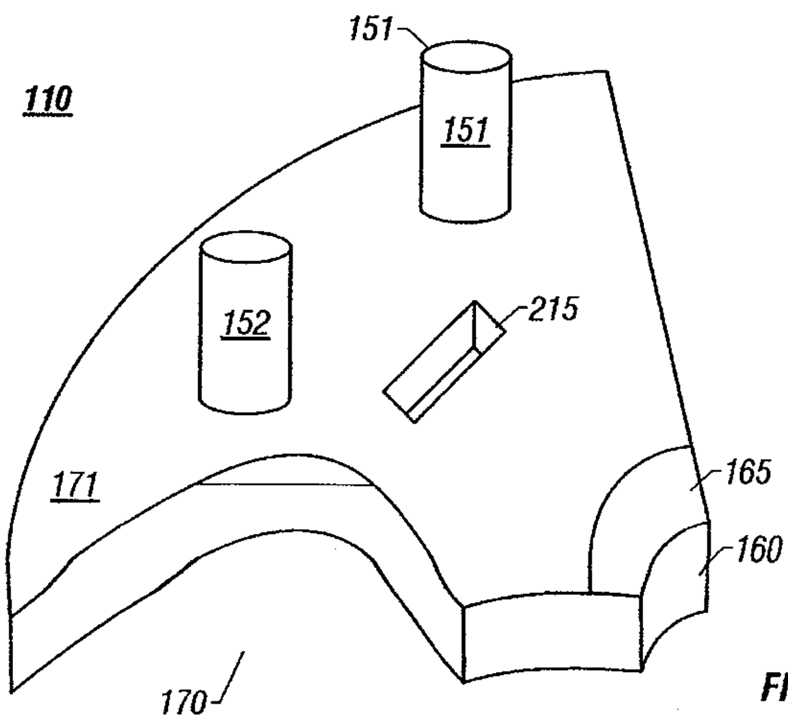


FIG. 1



2B
110
151
215
2B
152
160
A
156
170
150
210

FIG. 2A



151
110
151
152
215
171
165
160
170

FIG. 2B

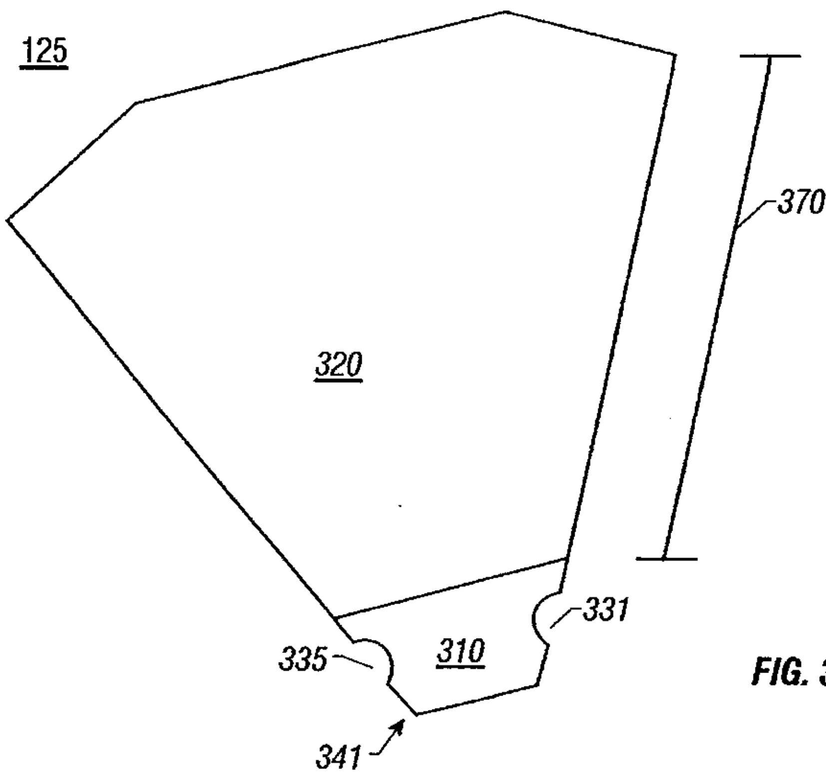


FIG. 3A

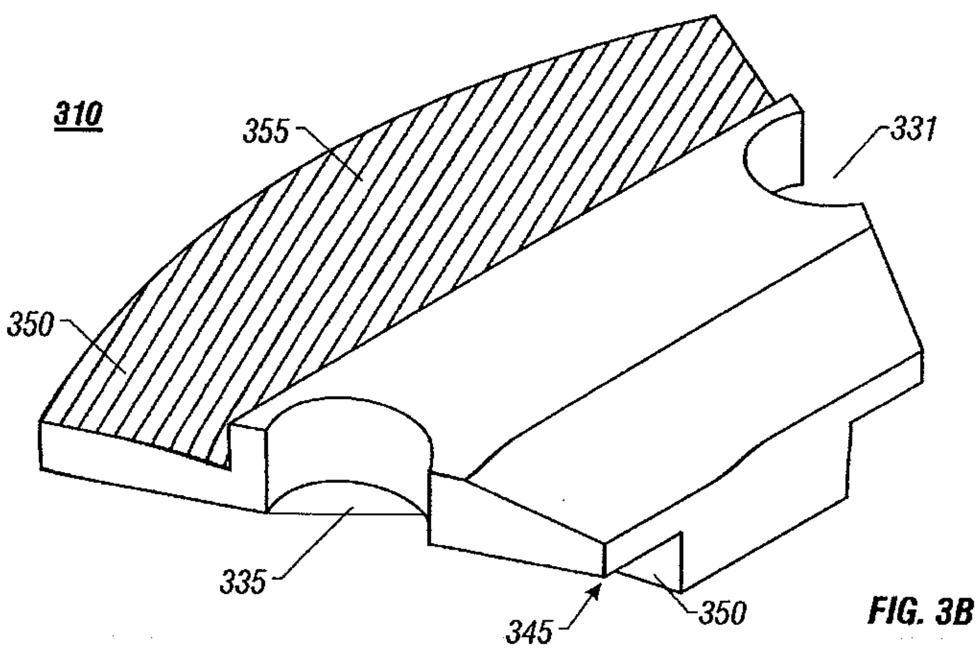


FIG. 3B

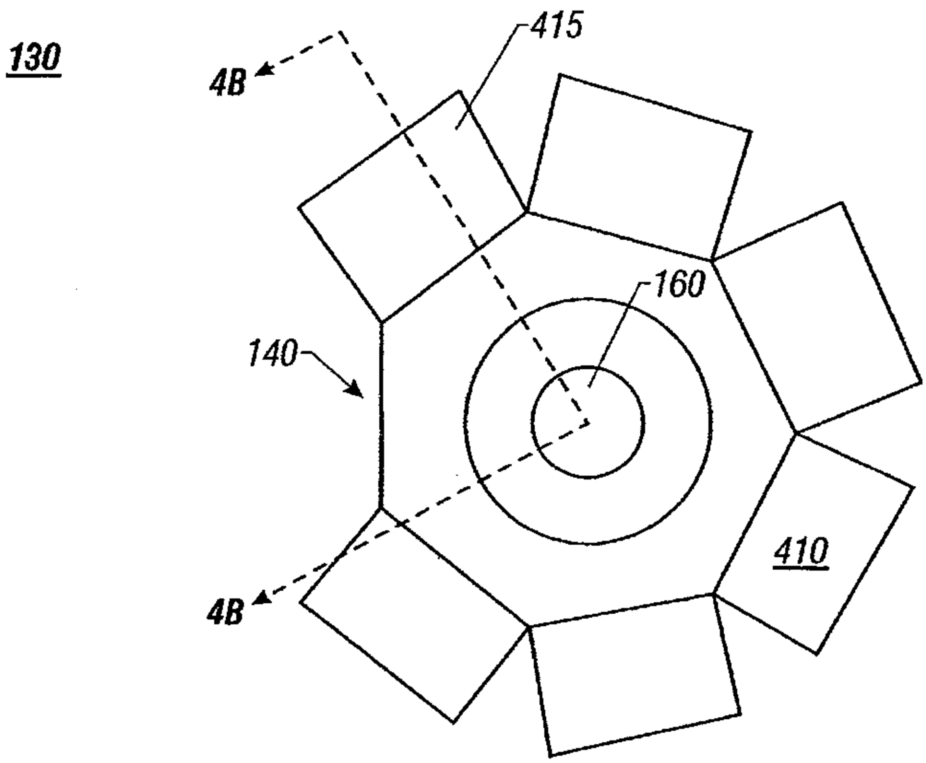


FIG. 4A

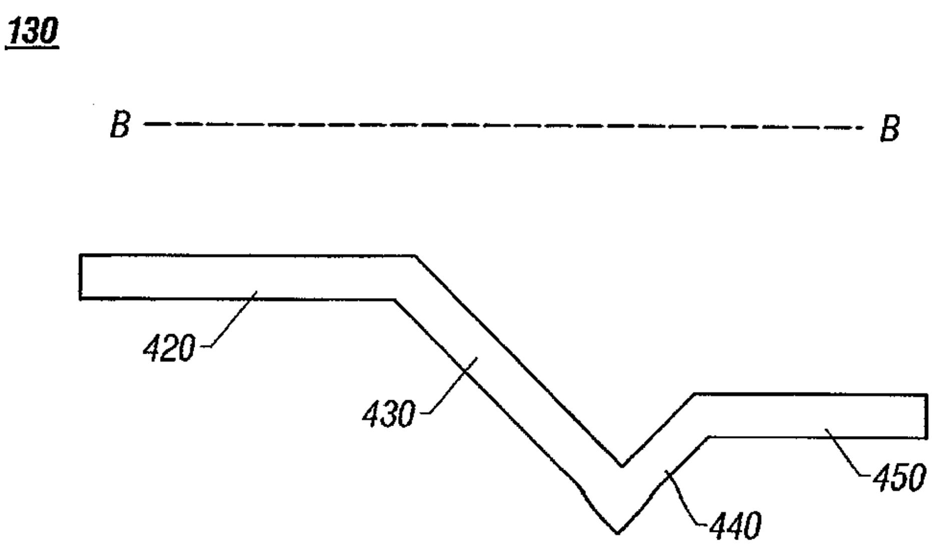


FIG. 4B

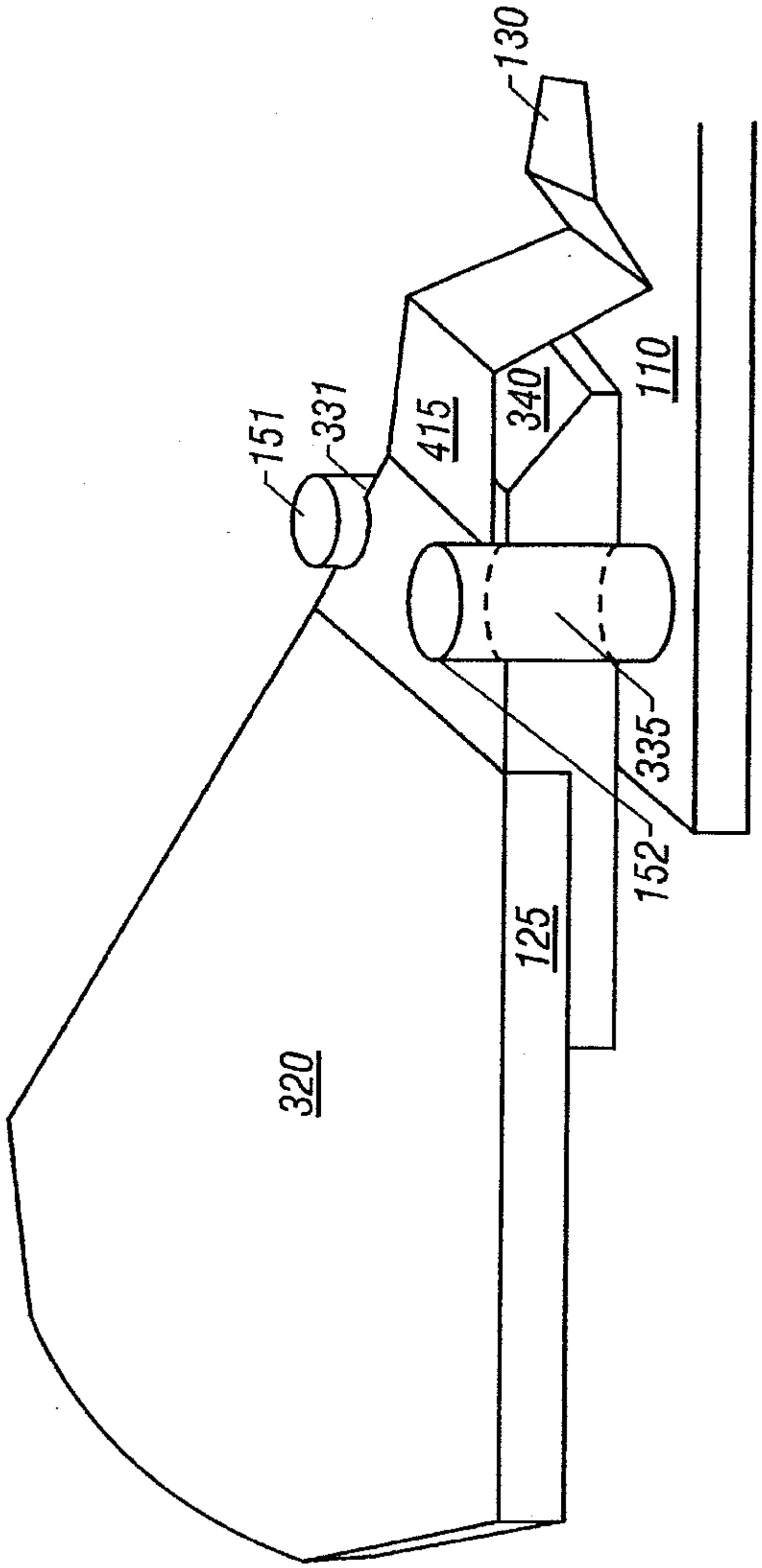


FIG. 5

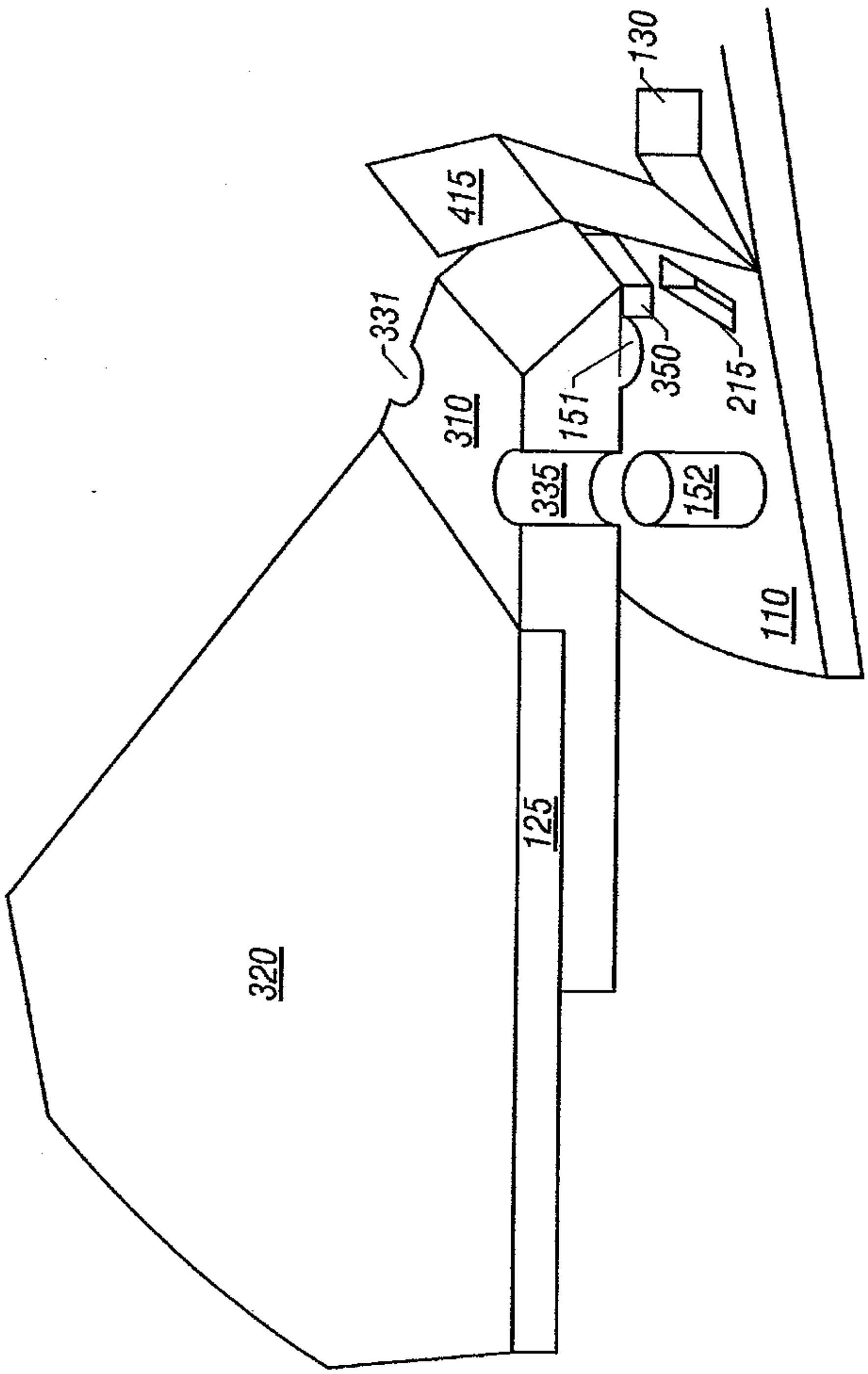


FIG. 6

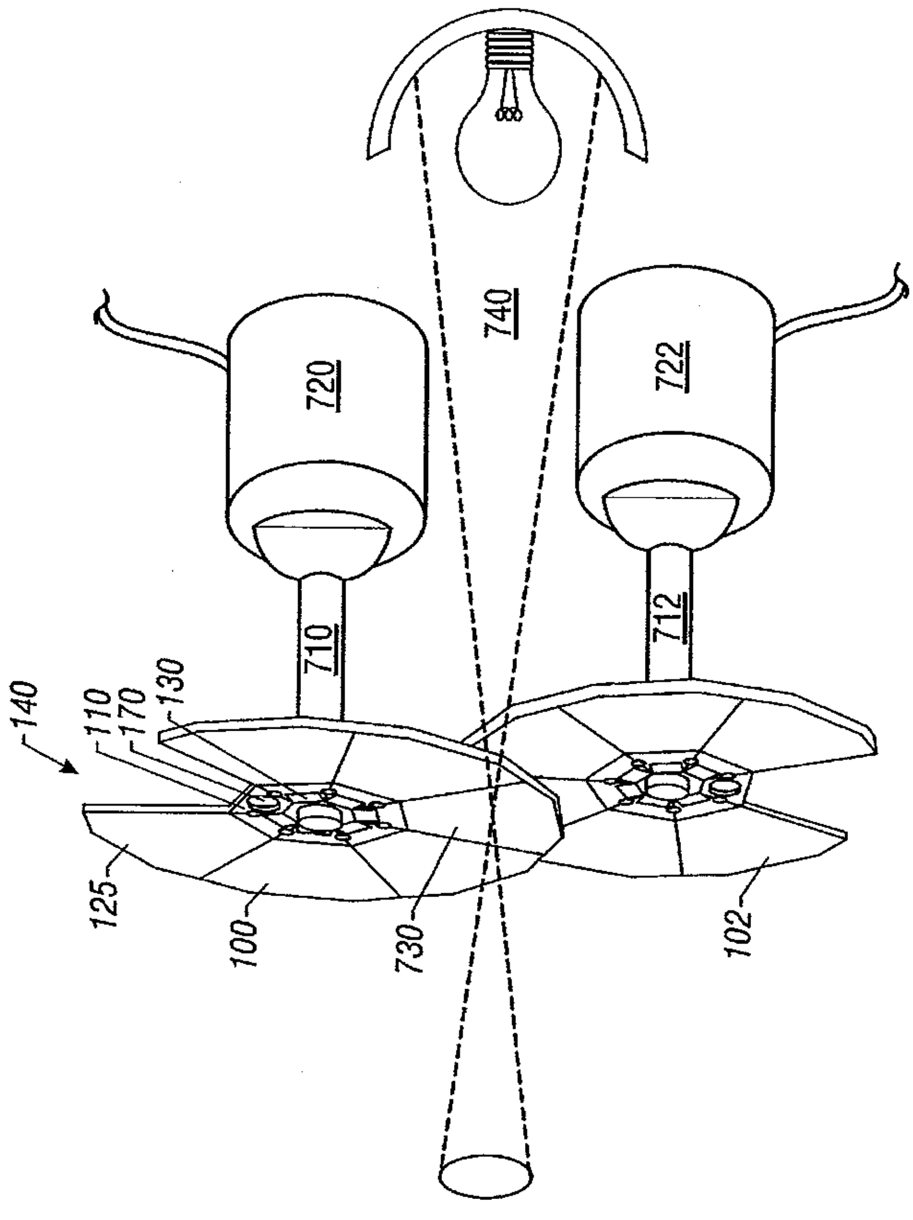


FIG. 7

CUSTOM COLOR WHEEL

FIELD

This disclosure relates to a rotatable color wheel for use with lighting equipment to produce multiple colors of light. More specifically, this disclosure provides a custom design color wheel with easily changeable, user-selected dichroic filters.

BACKGROUND

Some lighting equipment provides multiple colors by placing dichroic filters secured on a rotatable color wheel in the path of a light beam. Various color combinations can be produced by overlapping the filters.

The lighting fixture of U.S. Pat. No. 4,800,474 comprises two rotatable color wheels. The disclosure of this patent is herein incorporated by reference to the extent necessary for proper understanding. Each color wheel has a set of dichroic filters mounted about the periphery of a hub in which each of the filters in the set can be selectively positioned in a light beam by rotation of the color wheel. One of the color wheels is equipped with long wave pass dichroic filters while the other color wheel is equipped with short wave pass dichroic filters. By aligning various combinations of these filters, a number of different colors with different saturations can be produced.

The cutoff wavelengths for the dichroic filters are selected to be different at the long and short wavelengths for the filter set, such that there is a perceived uniform graduation of colors across the spectrum.

Each of the dichroic filters mounted on the color wheels is in a shape of a trapezoid and is mounted adjacent other filters, such that there is no blanking of light or leakage of light in the process of changing from one filter to the next.

This color wheel is constructed by joining two laminated aluminum plates. The diameter of the first plate is smaller than the diameter of the second plate. The difference in the diameters of the two plates form a step. The dichroic filters are bonded by an adhesive, preferably RTV silicon rubber, at the step. The dichroic filter is not easily removable after the filter is bonded to the plates by the adhesive.

The color wheel of U.S. Pat. No. 5,060,126 comprises a polygonal metal plate, which has slotted metal bars seated on the side edge of the polygonal plate with rectangular glass plates. One embodiment features dichroic filters adhesively held in the slots of the metal bars. The glass plates are not in direct contact with the metal bars and are spaced close to each other, but not in contact, in a peripheral array. The dichroic filters are not readily removed from the metal slots after the adhesive is applied.

The number of possible color combinations that can be produced by such a lighting fixture may be limited by the fixed number of secured filters on a particular color wheel. Some stage lighting applications may need different effects, and hence different sets of color filters during the course of a production. This may require custom color wheels.

SUMMARY

A color wheel is provided that has changeable filter elements. A hub has specific filter holding structures, including a spring element along the periphery of the hub.

Each color element has a color changing component mounted on a platform component. In a preferred embodiment, the platform component is shaped with concave edge elements, more preferably, half circles, that engage corresponding pegs on the hub, and a downwardly extending nub that engages with a corresponding cavity in the hub. The platform component has a beveled insertion edge on the top side.

The hub has a plurality of pegs and nub cavities. The color element is inserted into the hub and pressed down. The concave edge elements engage the pegs. The nub is inserted into the nub cavity. The color element is held in position on the hub by those structures as long as the bottom surface of the color element is pressed down against the hub. The spring element provides bias to do keep the element pressed down. The spring element extends just past the beveled edge on the platform component of the color element. The color element can be lifted from the hub by first lifting the spring element.

BRIEF DESCRIPTION OF THE DRAWING

These and other aspects will be described with reference to the drawings, in which:

FIG. 1 shows a top-view of a color wheel assembled with color elements held by a spring element;

FIG. 2A shows a top-view of a hub;

FIG. 2B shows a side-view of a hub;

FIG. 3A shows a top-view of a color element having a platform component and a transparent component;

FIG. 3B shows a three-dimensional view of a platform component;

FIG. 4A shows a top-view of a six-way retainer spring element;

FIG. 4B shows a cross-sectional view of a six-way retainer spring element;

FIG. 5 shows a side-view of an engaged color element in a hub held by a spring element;

FIG. 6 shows a side-view of a disengaged color element in relation to the hub and the spring element;

FIG. 7 shows a lighting device using a rotatable color wheel.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

FIG. 1 shows a top-view of a color wheel **100**.

The color wheel **100** is assembled with color changing elements **120** around the periphery of a hub **110**. One position **140** is preferably left open with no color elements attached. The color elements **120** are held in place when they are flat against the hub. Spring element **130** biases those elements **120** to be flat; and the spring force can be overcome to remove the spring elements.

The hub holds the color elements using a central nub, and two outer pegs. Each color element **125** is shaped to engage between two adjacent pegs **151, 152**. Each color element also has a downwardly projecting nub that extends into a corresponding nub recess in the hub.

The pegs **150** and nub cavity **215** are arranged generally into a triangular pattern for maximum holding.

The hub **110** also has a central opening **160**. In one embodiment, a collar **165** can be affixed to the hub **110**, surrounding the opening **160**. The collar **165** is formed to engage a shaft passing through the central opening **160** for rotation of the color wheel **110**. The spring element **130** is also adapted to allow a shaft to pass through a central opening **160**. The hub **110** and the spring element **130** are superimposed such that a shaft can pass through the central opening **160**.

A hole **170** is formed on the hub **110** at the open position **140**. ??WHAT IS THIS HOLE FOR?? This hole **170** on the hub **110** reduces the total weight of the color wheel **100**. Reducing the mass of the color wheel can be desirable. A reduced mass color wheel has a lower inertia which can facilitate operation during starting and stopping of the color wheel. However, the present inventors also anticipate an embodiment that comprises a color wheel without a hole **170**.

Each color element **120** is firmly held in the hub **110**. Once in place, color element **120** is held in position by the spring element **130**. The user can lift the spring element **130** to allow lifting the color element **120** and hence removing it.

FIG. **2A** shows a top-view of the hub **110** with no color elements **120** attached. A detailed perspective view of area A is shown in FIG. **2B**.

The hub **110** has a plurality of spaced pegs **150, 151, 152** and a plurality of nub cavities **210, 215**. Defining the surface seen in FIG. **2A** as the top, the pegs **150** stand vertically up from the top surface **171** of the hub **110**. The pegs have a rounded outer surface of diameter x, and a top surface **155**. The nub cavities **210** extend down into the hub **110**. The nub cavities **215** define a substantially rectangular inner surface, spaced between and laterally distant from pegs **151, 152**, thereby forming a triangular shape.

Each color element **125** on the color wheel **100** is held between the two pegs **151, 152**, and also held by the one nub cavity **215**.

FIG. **3A** shows a top-view of a color element **125** for use with the hub of FIGS. **2A–2B**. Each color element **125** has a platform component **310** holding a color filtering component **320**. The platform component **310** has concave openings **331, 335** that are shaped to engage between pegs **151, 152**, respectively, along each side holding surface **341**. The inner shape of each concave element **341** is substantially the same shape and size as the outer surface of the pegs. Each side holding surface **341** goes half way around the peg, and may also adjoin another side holding surface of a neighboring side holding surface.

FIG. **3B** shows a perspective view of one platform component **310**. The platform component **310** has a beveled edge **340** slanting toward the center of the color wheel **100**. A nub **350** is positioned on the bottom surface **345** of the beveled edge **340** and extends down farther than the otherwise flat bottom surface **345**. The nub **350** has a substantially rectangular shape with substantially the same outer dimensions as the outer dimensions of the nub cavity and fits into nub cavity **215** on the hub **110** as explained herein.

A descending step **350**, shown as the hatched area in FIG. **3B**, is formed at the other end of the platform. In one embodiment, the step **350** has a roughened surface **355** for improved adhesion with the transparent component **320**. The length **360** of the step **350** is selected to allow sufficient area for proper adhesion and support of the transparent component **320** on the color wheel **100**. One possible adhesion material for attaching dichroic filters is RTV silicon rubber, available from General Electric and DuPont.

The transparent component **320** can be conventional colored glass or dichroic filters. Dichroic filters are well known in the art. Dichroic filters can be formed from a sheet of thermally resistant or refractory glass, e.g. PYREX™ or quartz. The sheet is coated to separate two pre-selected, complementary colors out of white light, one of which is reflected, and one transmitted.

FIG. **4A** shows a six-way retainer spring element **130**. In an embodiment with six color elements on a color wheel, a six-way retainer spring element is used. Similarly, for an embodiment with n-color elements, an n-way retainer spring element can be used.

The spring element **130** biases the platform element **310** into its lowermost position, such that bottom surface **345** is pressed against top surface **171** of the hub. While pressed down like this, the color element **125** is maintained firmly in place between the vertical pegs **151,152**, and in the nub cavity **215** on the hub **110**.

The spring element **130** has a central opening **160**, provided to allow a shaft to pass through. The six-way retainer spring element **130** has a plurality of arms **410**. No arm is positioned at the open position **140**.

FIG. **4B** shows the cross-section B—B of the spring element **130**. The cross-section of the spring element **130** has a horizontal segment **420**, a downward sloping segment **430**, a slight upward sloping segment **440**, and a small horizonal segment **450**.

FIG. **5** is a side-view of an engaged color element **125** held in position by the spring element **130** in the hub **110**. Each arm **415** of the spring element **130** extends just past the beveled edge **340** of the color element **125**. Preferably, the arm **415** reaches the part of the platform **310** before the concave openings **331, 335** begin. The arm **415** does not extend past the platform component **310**; the spring element **130** does not touch the transparent component **320** of the color element **125**.

The color element **125** clicks into the hub **110**. The concave openings **331, 335** engage the pegs **151, 152** respectively. The nub **350** fits into the nub cavity **215**. When engaged, the nub **350** is not visible from the side-view as in FIG. **5** and the bottom surface **345** is pressed against the top surface **171** of the hub. The position of the color element **125** is securely maintained by the triangular layout of the supports. The supports extend in opposing directions, in the sense that two supports extend upward and one support extends downward.

The color element **125** can be removed from the hub **110** by lifting the spring element **130** as shown in FIG. **6**. Since the spring element **130** extends just past the beveled edge **340**, users may slide their fingers under the spring element **130** to lift the spring element **130**. After the spring element **130** is lifted, the entire color element **125** can be lifted vertically from the hub **110**. The filter element needs to be lifted sufficiently high that the bottom surface **145** is above the top surface **155** of the pegs **151, 152**, and the nub is completely out of the hub recess. Then the color element **125** is in the position shown in FIG. **6**, where it can be removed. The opposite operation, thereby lifting the spring, is carried out to insert a new color element.

FIG. **7** shows one embodiment using the color wheel **100**. Color wheels **100, 102** are adapted to engage shafts **710, 712**, respectively. The color wheels **100, 102** are each rotated by a motor **720, 722** respectively. The color wheels **100, 102** are positioned such that the two color wheels **100, 102** overlap at one filter position **730**. Filter position **730** is in the path of a beam of light **740**.

Although only a few embodiments have been described in detail above, those having ordinary skill in the art will certainly understand that many modifications are possible in the preferred embodiment without departing from the teachings thereof. Other materials can be used to fabricate each part of the color wheel. The hub can be made without the hole **170**. The platform can be made without a roughened surface **355** on the step **350**. A plurality of color wheels can be incorporated in a lighting apparatus. Moreover, the nubs and pegs can be reversed or formed into different geometric shapes.

While three support structures—two pegs and a nub—are preferred in the form described herein, any arrangement or kind of structure could be used. At least one, more preferably two, and most preferably three structures in a triangular shape are contemplated. Moreover, the gender of the support structures can be reversed from that described herein—with pegs and cavities being switched.

All such modifications are intended to be encompassed within the following claims.

What is claimed is:

1. A color filter comprising:

a hub having a top surface having at least one first shaped peg which extends above the top surface and at least one second shaped nub cavity extending below said first surface;

at least one color element, having a bottom surface, and having at least one peg accepting surface, shaped to engage with at least a portion and said peg, and at least one nub, shaped to engage with at least a portion of said nub cavity on said hub; and

a spring element, positioned to hold at least one color element in place on said hub.

2. A color filter as in claim 1, wherein said at least one color element is pressed down to engage and lifted to remove.

3. A color filter as in claim 1, wherein there are a total of three of said pegs and nub cavities on said hub, arranged generally into a triangular shape.

4. A color filter as in claim 3, wherein said pegs are cylindrical and said nub cavities are rectangular.

5. A color filter as in claim 2, wherein said spring includes a lifting surface, facilitating its lifting so that said color element can be lifted.

6. A filter as in claim 1, wherein there are two of said pegs and said color elements are shaped with peg accepting surfaces at side surfaces thereof.

7. A color filter as in claim 6, wherein said pegs are cylindrical.

8. A color filter as in claim 7, wherein said accepting surfaces are the shape of half circles.

9. A removable assembly comprising:

a platform component, formed with a top surface with a beveled edge, a bottom surface with an extending nub and shaped side surfaces formed into a predetermined shape, and a filter attachment portion having a surface to which a filter can be attached, said attachment portion having a first length; and

a hub portion, having a nub cavity shaped to accept said nub of said platform component, and having extending pegs, shaped to interact with said shaped side surfaces of said platform component.

10. An assembly as in claim 1, further comprising a transparent component fixably supported on said filter attachment portion of said platform component, wherein said transparent component has a second length, and said second length is longer than said first length.

11. An assembly as in claim 9, wherein said filter attachment portion has a roughened surface formed to facilitate adhesion of an object.

12. An assembly as in claim 9, wherein said pegs are cylindrical.

13. A removable color filter assembly comprising:

a hub having a support surface and at least one first gender supporting structure;

a color filter holding assembly, including a color filtering element, and having a second gender supporting structure, said second gender supporting structure interacting with said first gender supporting structure; and

a removable holding element, holding said color filter holding assembly relative to said hub when in a first position, and allowing release of said color filter holding assembly relative to said hub in a second position.

14. An assembly as in claim 13, wherein there are two of each of said first and second gender supporting structures.

15. An assembly as in claim 13, wherein there are three of each of said first and second gender supporting structures.

16. An assembly as in claim 15, wherein said supporting structures are arranged in a triangular shape.

17. An assembly as in claim 13, wherein said supporting structures comprise a plurality of pegs and nub cavities formed on said support layer, wherein said pegs protrude from said support layer and said nub cavities are hollowed into said support layer.

18. An assembly as in claim 13, wherein said removable holding element comprises a spring element that has:

a horizontal segment having a first length;

a downward sloping segment having a second length;

a upward sloping segment having a third length;

a second horizontal segment having a fourth length,

wherein said first and second lengths are greater than said third and fourth lengths.

19. An assembly as in claim 13, wherein said holding assembly has a beveled surface, and said holding element comprises:

a spring component engageable with said beveled surface, wherein when engaged, said spring component holds said beveled structure in place, said component further allows said beveled structure to be lifted when said component is lifted from the surface of the beveled structure.

20. A method of engaging selectively removable color elements on a color wheel comprising:

obtaining a support surface having a plurality of support structures, including at least one protruding peg and a cavity;

obtaining at least one color element having a protruding nub shaped to interact with and enter said cavity, and a peg interacting surface shaped to engage with said protruding peg;

engaging said color element with said support surface such that said nub is within said cavity, and said peg interacting surface is interacting with said peg;

holding said color element in place on said support surface when desired; and

allowing said element to be lifted such that a bottom surface of said nub is above said support surface for removal of said color element.

* * * * *

EXHIBIT G

US006029122A

United States Patent [19]

Hunt

[11] Patent Number: 6,029,122

[45] Date of Patent: Feb. 22, 2000

[54] **TEMPO SYNCHRONIZATION SYSTEM FOR A MOVING LIGHT ASSEMBLY**

[75] Inventor: **Mark A. Hunt**, Birmingham, United Kingdom

[73] Assignee: **Light & Sound Design, Ltd.**, Birmingham, United Kingdom

[21] Appl. No.: **09/034,045**

[22] Filed: **Mar. 3, 1998**

Related U.S. Application Data

[60] Provisional application No. 60/038,136, Mar. 3, 1997.

[51] **Int. Cl.**[7] **F21V 33/00**

[52] **U.S. Cl.** **702/188**; 364/132; 362/85; 340/825.5

[58] **Field of Search** 702/188; 364/132; 362/85, 233, 239; 340/825.5, 825.49

[56] **References Cited**

U.S. PATENT DOCUMENTS

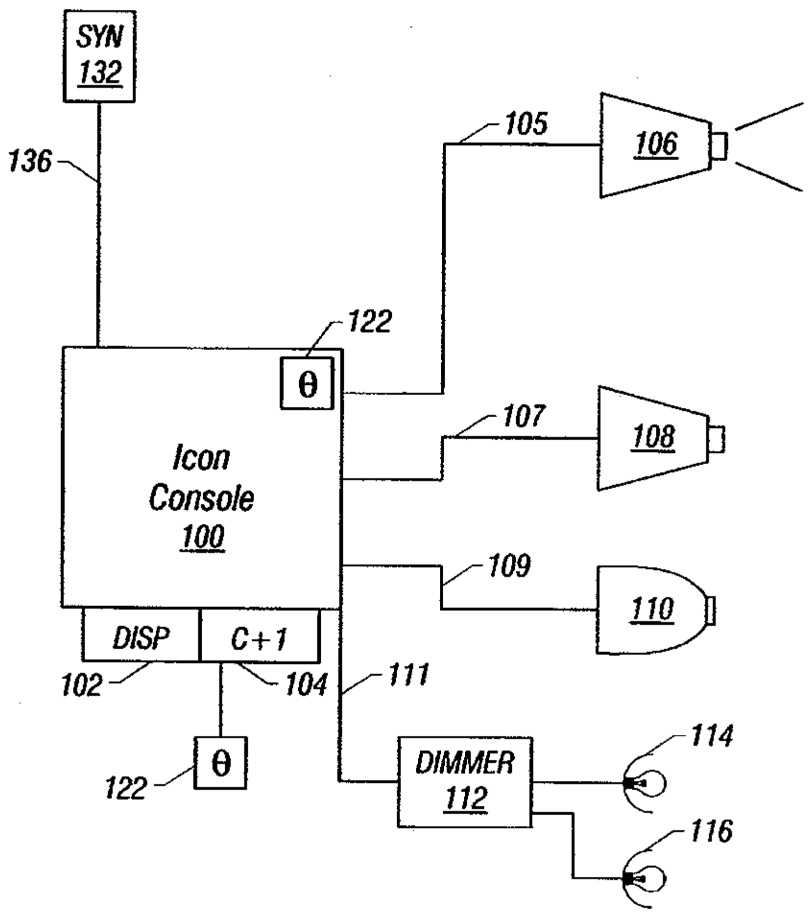
4,419,918	12/1983	Dyck et al.	364/761
4,980,806	12/1990	Taylor et al.	364/132
5,406,176	4/1995	Sugden	362/233
5,504,477	4/1996	Whitright et al.	340/825.5
5,769,527	6/1998	Taylor et al.	362/85

Primary Examiner—Marc S. Hoff
Assistant Examiner—Hien Vo
Attorney, Agent, or Firm—Fish & Richardson P.C.

[57] **ABSTRACT**

A time synchronizing unit operates to synchronize lamps with various indicia. Each of these indicia are tied to a system clock. The system clock is normally incremented at specified intervals, e.g., every 4 milliseconds. The system is to be synchronized with a stage lighting production such as a song. If the tempo of the song changes, the amount of time between interrupts changes. Hence the time when things occur will be varied as the system clock changes. This enables varying the time when the things occur without actually changing the program: the time is bent to accommodate the new tempo of the program.

25 Claims, 4 Drawing Sheets



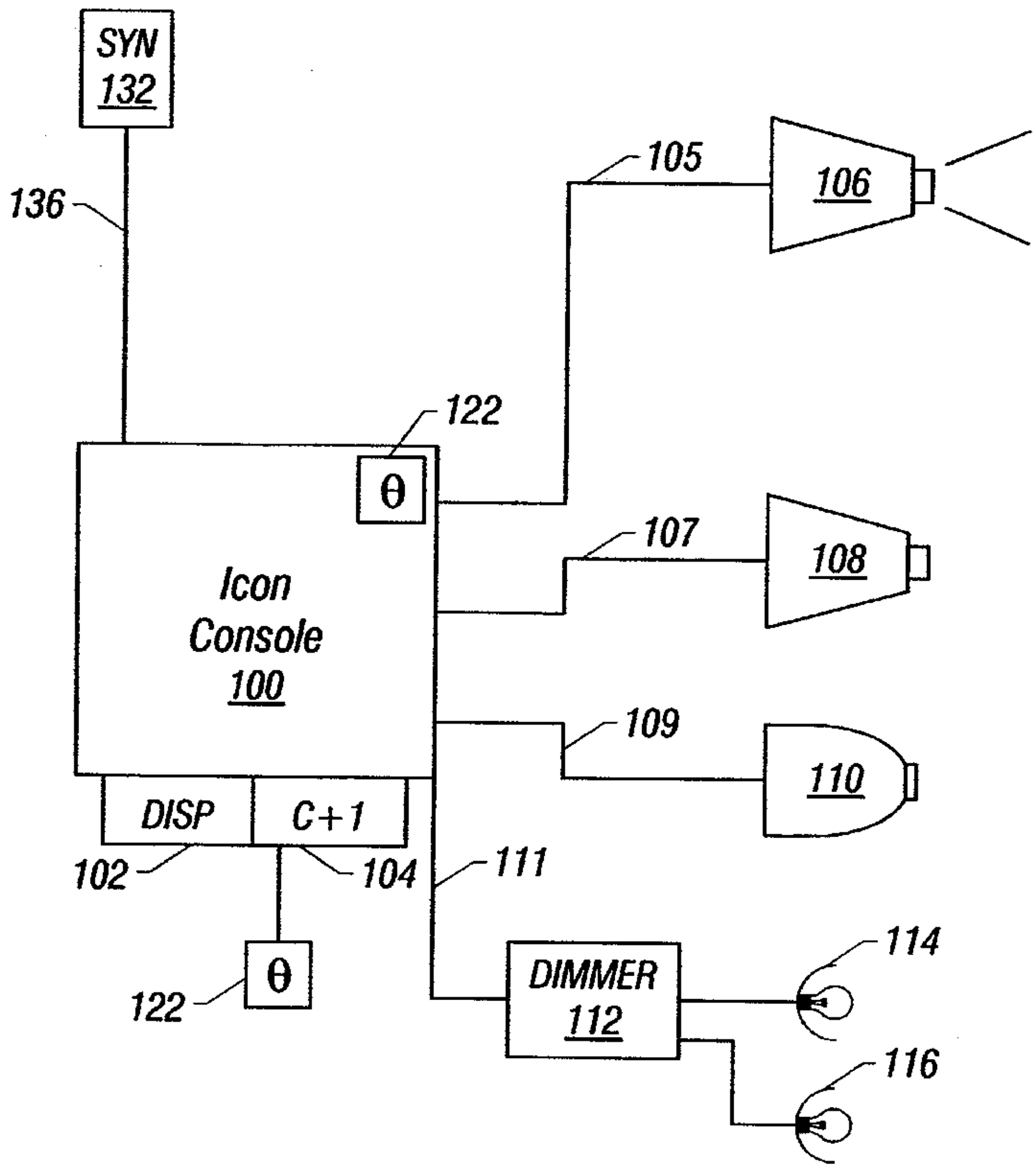
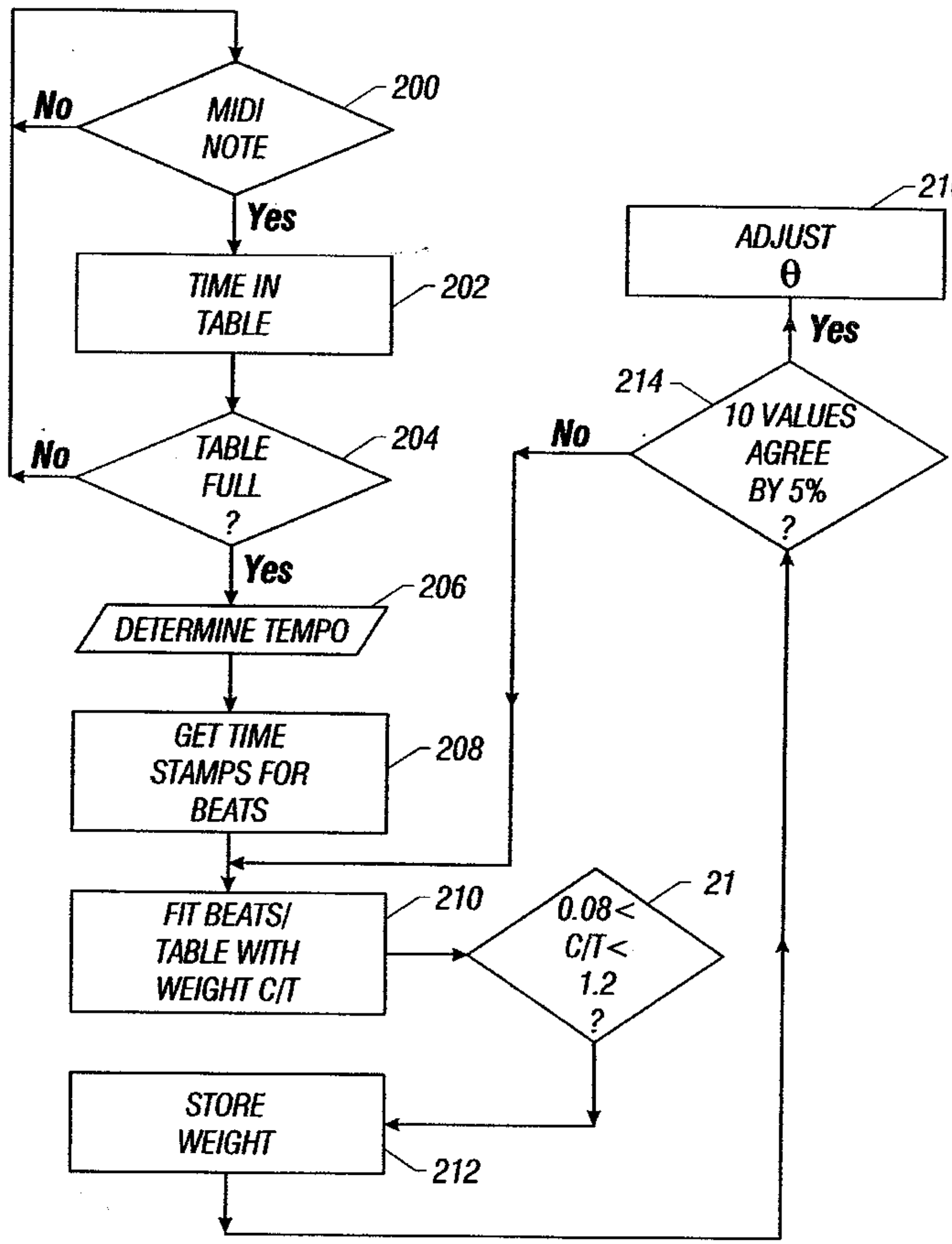

SYN
132
136
105
106
122
θ
107
108
Icon
Console
100
109
110
DISP
C+1
102
104
111
θ
122
DIMMER
112
114
116

FIG. 1



MIDI NOTE
200
No
Yes
TIME IN TABLE
202
TABLE FULL ?
204
No
Yes
DETERMINE TEMPO
206
GET TIME STAMPS FOR BEATS
208
FIT BEATS/ TABLE WITH WEIGHT C/T
210
0.08< C/T< 1.2 ?
21
STORE WEIGHT
212
10 VALUES AGREE BY 5% ?
214
No
Yes
ADJUST θ
216

FIG. 2

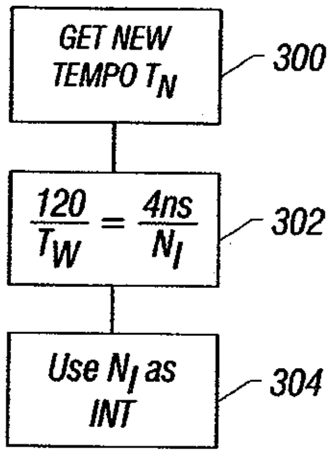

GET NEW TEMPO T_N
300
$\frac{120}{T_W} = \frac{4ns}{N_I}$
302
Use N_I as INT
304

FIG. 3

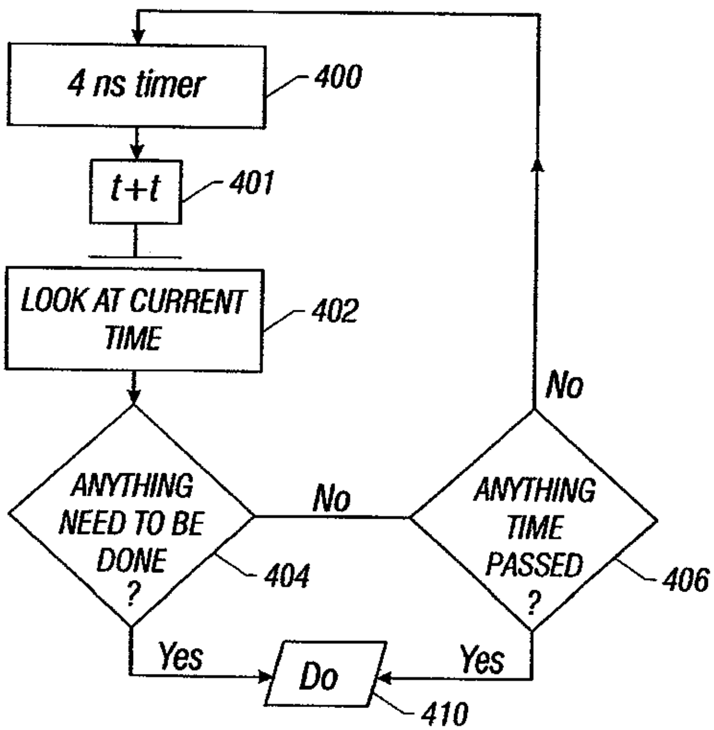

4 ns timer
400
t+t
401
LOOK AT CURRENT TIME
402
ANYTHING NEED TO BE DONE ?
404
No
ANYTHING TIME PASSED ?
406
No
Yes
Do
Yes
410

FIG. 4

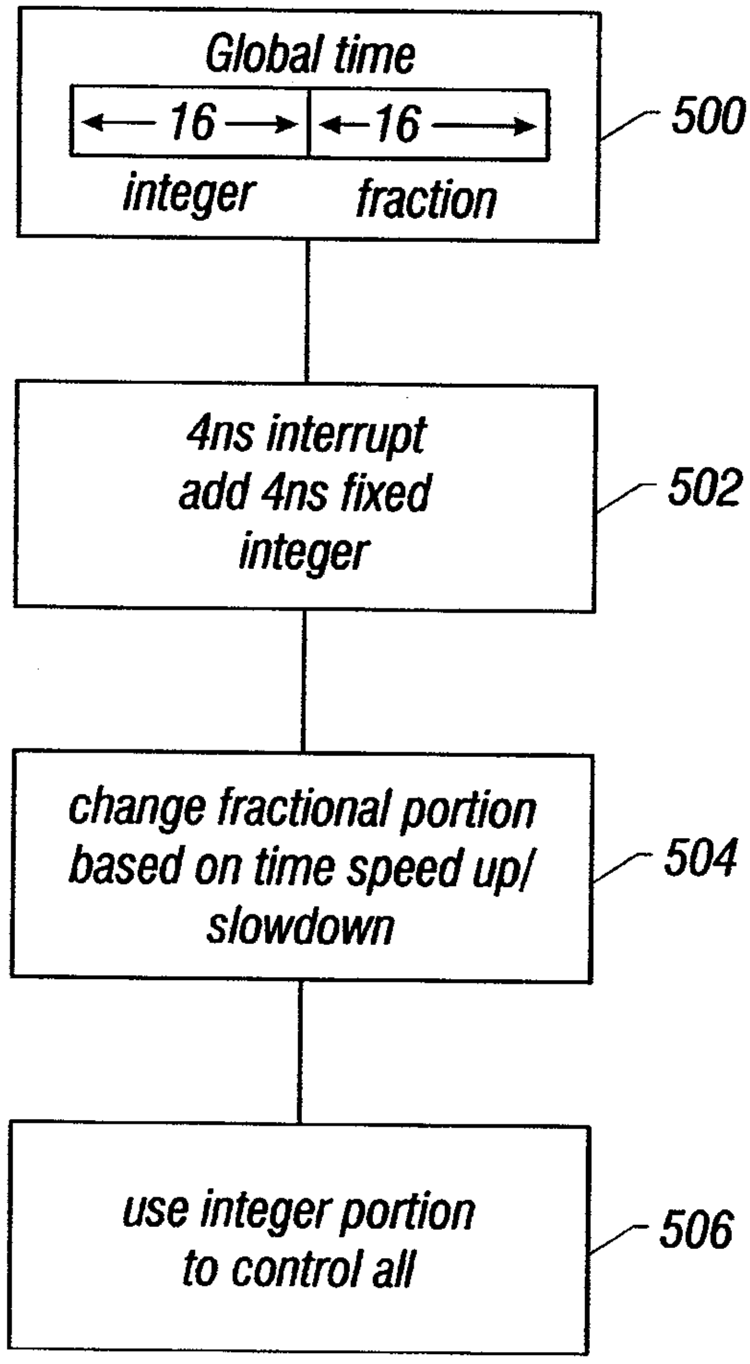

Global time
16
16
integer
fraction
500
4ns interrupt
add 4ns fixed
integer
502
change fractional portion
based on time speed up/
slowdown
504
use integer portion
to control all
506

FIG. 5

TEMPO SYNCHRONIZATION SYSTEM FOR A MOVING LIGHT ASSEMBLY

This application claims the benefit of U.S. Provisional application Ser. No. 60/038,136 filed Mar. 3, 1997.

FIELD OF THE INVENTION

This application relates to a tempo synchronization system for a stage lighting system. More specifically, the present application describes a system which enables global adjustment of clock values in a plurality of remotely controlled stage lighting units, and synchronizes all of those stage lighting units with a common variable source.

BACKGROUND AND SUMMARY

Modern stage lighting systems are typically controlled by computer-based controlling systems.

The complexity of such stage lighting systems has necessitated increased computing power. There has been a tendency to distribute that increased computing power among the luminaires by making each of the luminaires into an intelligent subsystem.

The distributed computing power is effected by a central processing computer and a number of slave processing computers which accept their commands from the central processing computer.

One example of such a system is the ICON(TM) system made by Light & Sound Design, Ltd., the assignee of the present application. The Light and Sound system uses a central controller ("the ICON controller") to control a plurality of distributed processing units that are located in the ICON lamp units. Each ICON lamp unit has a number of processors controlling various functions of the unit.

An exemplary command from the ICON controller to the ICON unit might be parsed as: go to position X at speed S and be there at time Y. The ICON unit receives this command, and uses its own processor to determine how to drive the motors and when to start, etc, to carry out this command.

According to the preferred embodiment, all of the operations are synchronized to a single clock: a system clock. This has the advantage of requiring maintenance of only one clock.

The inventors recognized that various things, including timing, may change during a performance. Sometimes the tempo of the song they are playing may change between rehearsal and the playing time. Some aspects of the lighting performance may need to be synchronized with that operation.

The inventors of the present invention, having recognized these problems, have devised a feature allowing the timing of operations to be changed globally. A preferred embodiment describes a system where operations of both the main system and the subsystem are tied to the system clock. The system clock can be selectively changed in a way that causes those synchronized operations to operate at a different effective rate. When the time for one clock pulse changes, everything referenced to that clock pulse will change as well. This enables, for example, the tempo provided by the clock to be changed to match the tempo of a particular song being played.

As an example of the way the system could be used, assume a light chase including a plurality of lighting effects. Lighting effect no. 2 follows lighting effect no. 1 and lighting effect no. 3 follows lighting effect no. 2 and so on. This all continues until the end of the chase. The chases can be synchronized with, for example, clock timings.

A system clock is changed in order to change various aspects which are synchronized with that system clock. According to one aspect, a show with sequential lighting events has operations which are timed to coincide with incidents in music. The cues are triggered from a system timing element in the console, and times are controlled by the clock in the console. According to this aspect, the console's effective clock running frequency is modified. The cues remain synchronized with the clock, but since the clock has been changed, the cues occur at different absolute times. This operation is referred to herein as Time Bending (TM).

Specific ways of carrying out this operation are described.

BRIEF DESCRIPTION OF THE DRAWINGS

These and other aspects will now be described in detail with reference to the accompanying drawings, wherein:

FIG. **1** shows a block diagram of the invention;

FIG. **2** shows a flowchart of an automated operation of this system;

FIG. **3** shows a flowchart of how the tempo is used to adjust timings;

FIG. **4** shows a flowchart illustrating how operations are carried out relative to timing; and

FIG. **5** shows a flowchart of operation showing maintaining global time as an integer part and a fractional part.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

FIG. **1** shows a block diagram of the preferred embodiment using a time bending system. ICON console **100** includes processor circuitry therein including a clock element **120** which is preferably effected as described herein. ICON console **100** also includes a display unit **102** and a controlling unit **104**.

The architecture of the ICON console uses a separate controlling line for each multi-parameter fixture being controlled. FIG. **1** includes two ICON units **106**, **108**, although in reality many more would be expected. A Wash Lamp (TM) unit **110** is also shown. Single parameter fixtures **114**, **116**, may be controlled from a single line **111**.

Therefore, lines **105**, **107**, and **109** each carry information from a single unit. Line **111**, however, may carry a plurality of time division multiplexed information. This information is distributed to dimmer rack **112** which feeds a number of single parameter (e.g., dim-only) lighting fixtures **116**, **118**.

Both the ICON console **100** and the ICON units **106**, **108** include their own processor controlling the operations. The Wash Lamp unit **110** may also include a processor. These elements of the lighting system are commanded at certain times to carry out a prestored "cue", which can be either an operation or a sequence of various operations.

A show is formed of a sequence of lighting events which are carried out to occur one after another within the show. At least some of the cues are timed to coincide with incidents within the music being played. However, those cues are actually triggered by the clock in the console. Hence, those timings within the system are controlled by the clock in the console. The cue is commanded to occur at the right moment. The timing of the cue is often determined by incidents in the music.

No two shows, however, have precisely the same tempo. Therefore, the inventors have devised a way of changing the

entire timing of the system. This is done by adjusting the console's concept of time.

Time is measured by beats per minute, i.e., the number of beats which occur in one minute. The ICON usually uses 120 beats per minute, and the timing of the shows is based on that 120 beats per minute. According to a first aspect of the present invention, a new time can be entered by encoder knob **122**. This entry device can be an optical encoder which detects movements and produces pulses corresponding to those movements.

This embodiment uses a clock which pulses at 120 beats per minute. The encoder commands change of the speed of the clock. The encoder includes controls between +10 and −10. Positive 10 in this embodiment commands an increase of speed of the clock by 50% to 180 beats per minute, while negative 10 can represent a slowdown of the clock by 50% to 60 beats per minute. An alternative system simply uses each predetermined amount of movement of the encoder representing 5 beats per minute. Of course, any other scale and amount of rotation could alternately be allowed.

The actual clock value is displayed on display **102**.

A second technique allows the current tempo to be entered via control **104**. Control system **104** allows entry of an "edit time" command. When the edit time button is pressed, the console enters a data entry mode. A new tempo in beats per minute can be entered into the system.

Yet another system of controlling tempo uses an automated approach which synchronizes with the music being played. The ICON console **100** receives a MIDI input over cable **130** from one or more musical instruments. For example, synthesizer **132** produces a MIDI output which is connected to ICON console **100**. In this embodiment, ICON console monitors the MIDI output to determine a tempo from that MIDI stream.

The flowchart of FIG. **2** is carried out for each song which is played. At step **200**, the system monitors for the presence of a MIDI event representing a note having been played. If such a MIDI event is not received at **200**, the system continues to monitor the MIDI note stream until a note is detected.

The embodiment takes cognizance of the fact that not every note will be played at every tempo period. Therefore, an expected fit of synchronization with these MIDI notes needs to be made.

A MIDI note is detected at step **200**. The time when the MIDI note was produced is stored as a time stamp in a table at **202**. A predetermined number of samples being stored in the table at **202** represents a statistical sample that is likely to allow determination of tempo, e.g. 300 samples. When the table is considered as full at step **204**, control passes to step **206** which determines the tempo from the values in the table.

The preferred determination of tempo value is based on the assumption that the tempo value already being used is close to the actual tempo value.

This system calculates times between the time stamps at step **208**.

Synchronization is carried out at step **210** by calculating the weighting value C/T, where C is the time between beats of the clock, and T is the time between time stamps determined at step **208**. Only C/T values which are close to 1 are retained; the others are assumed to be odd notes. If C/T is determined to be between 0.8 and 1.2 at step **211**, the value is stored into a table at step **212** as a weighting value. When ten values are determined to agree by 5% at step **214**, that weighting value is used to adjust the clock C at **216**.

The value of weighting which corresponds to the closest fit is then used to adjust the tempo. When the tempo is adjusted by that weighted value, the new tempo becomes the closest fit to the musical events.

Once the new tempo has been determined by any of the above techniques, (encoder, edit time, or MIDI autodetect) the time in the console is changed in accordance with that new tempo. This is carried out according to the flowchart of FIG. **3**.

Step **300** represents the system acquiring the new tempo using one of the techniques described previously, or any other technique of obtaining a new tempo for the entire system. That tempo needs to be translated into the console at step **301**.

The ICON console operates using a processor which is clocked at 16 megahertz. Tempo is controlled according to the flowchart of FIG. **4**. Every predetermined period of time P, nominally 4 ms, a timer produces a special pulse at step **400** which initiates a timer determination routine which is diagramed in FIG. **4**.

At step **401**, the system timer is incremented so that the system time stamp is increased. This new incremented time will be used during the next interrupt. The current time is noted at step **402**. At step **404** the system determines from the current time whether anything needs to be done. For example, at a predetermined time during the show, a cue may need to be advanced or a light moved. Each time the period P elapses, the system determines whether any of those timed events need to be executed.

If nothing needs to be done at step **404**, the system determines at step **406** whether there is anything that needs to be done whose time has passed and is not yet done. If not, the interrupt ends and the routine is again executed in another time T. If either steps **404** or **406** are positive, however, control passes to step **410** which commands that the operation be effected.

This embodiment modifies the timer by adjusting the time between interrupts. This embodiment uses a time interrupt every T ms which increments the timer. Therefore each T ms advances the time by a fixed amount: T ms at 120 beats per minute. This is actually a granular clock speed adjustment, but is sufficiently fast that human perception cannot determine that things are occurring with granularity. Accordingly, this system determines at each interrupt if something needs to be done and does it if necessary.

Therefore, returning to the flowchart of FIG. **3**, after obtaining the new tempo T_n, the system determines a new interrupt time from the tempo according to the relation

$$T_{new} = \frac{t_n}{C} \cdot T$$

where C is the pulses per second of the clock, T is the nominal interrupt value. Here C=120 and T=4 ms. This is carried out at step **302**.

At step **304**, the new tempo is written to the software timer which carries out the interrupt.

This has an important advantage in that when the tempo changes, the programming which has already been done does not need to be changed. Everything carries on as usual, since the programming still occurs at predetermined clock times. The clock's knowledge of time is changed: not the programming.

Yet another embodiment of the present invention uses the techniques shown with reference to the flowchart of FIG. **5**. Step **500** shows the global time variable form in which time

is maintained. The global time variable includes 32 bits: an integer 16-bit portion and a fractional 16-bit portion. The entire integer/fraction portion is used to maintain the time of the system. However, the timed elements respond only to the 16-bit integer part.

Step **502** represents each 4 ms interrupt period. A fixed point number is added to the interrupt routine. This fixed point portion is equivalent to 4 ms. Step **504** changes the fractional portion based on the time speedup/showdown. Hence, this changes the fractional portion according to that amount.

As explained above, the integer portion is used at step **506** to control all controlled features in the system. However, the main timing system maintains the fraction. The fraction, through its overflow/underflow effect on the integer, has the effect of granularly changing the timing operation.

All of the above systems have described use for global time maintaining where this global time maintaining operation can be local or remote. In a remote embodiment, either the time can be transferred to each remotely controlled device such as, remotely controlled luminaries, or a number can be transferred to those remotely controlled devices to change remote clocks throughout the system in an analogous way.

While the above has described using a software timer as the interrupt, it should be understood that the 16 MHZ clock could also be divided using a down counter operation. For example, the down counter can be preset with a particular value, and then initiated to count down from that particular value using the 16 MHZ clock. Each time the counter counts down to 0, this produces a clock pulse which also resets the counter, and re-presets the count down value. Different values being preset into the down counter allow different granularities of the clock with a 1/16 MHZ granularity rate.

The operations carried on by the processor using this system are not in general changed by this system. The third embodiment described herein actually changes the processor clocking frequency. This changes the speed at which the processor operates. Many processors can be overclocked and underclocked in this way, while the system still operates properly. This requires that the processing unit be capable of operating at various clocking speeds. In contrast, the system of this first embodiment does not change anything about the way the processor operates. It only changes the time for purposes of controlling operations, without changing the processing speed of the clock.

This modification uses this system to actually adjust the clock for the microprocessor itself. In this case, the microprocessor clock of 16 megahertz might be varied between 14 megahertz and 18 megahertz. This could also be done using a fast clock such as a 100 megahertz clock which is divided down to form the microprocessor clock. This would have the same effect on the 4 ms interrupt: the system granularity would change in proportion to the change of clock. Applicants also contemplate that this system could be used without the 4 ms granularity described above, especially if the processor speed itself was changed as described in this modified embodiment.

Although only a few embodiments have been described in detail above, those having ordinary skill in the art will certainly understand that many modifications are possible in the preferred embodiment without departing from the teachings thereof. For example, it should be understood that while the present system describes single line per channel control, the present inventors also contemplate using the present invention with a system that controls a plurality of multi-parameter lights over a single line. Examples of such systems are found in systems by Vari lite, Inc. (TM) and systems controlled according to the industry standard DMX 512 protocol.

All such modifications are intended to be encompassed within the following claims.

What is claimed is:

1. A distributed stage lighting system with adjustable tempo, comprising:

a plurality of controlled stage lights, distributed at a plurality of locations; and

a controller for said stage lights, said controller connected to said stage lights and including a system clock that maintains an indication of time;

said controller including a commanding element which sends commands to said plurality of lights at predetermined times that are set by the system clock in the controller and a time varying element which varies said indication of time in a way that causes said predetermined times of operation of said plurality of lights to be varied relative to one another.

2. A system as in claim **1** wherein said controller uses a system which checks at every predetermined interval of time to determine if an action should be taken.

3. A system as in claim **2** wherein said time varying element changes said interval of time to thereby adjust an amount of granularity between actions.

4. The system as in claim **1** wherein said time varying element varies a system clock.

5. The system as in claim **4** wherein said system clock is a clock of the microprocessor.

6. The system as in claim **4** wherein the said system clock is an incremented clock where incrementing occurs every predetermined interval of time, and wherein said interval of time is adjusted by said time varying unit.

7. The system as in claim **2**, wherein said system includes interrupt driven routines, one of said routines incrementing a clock value by a specified amount, and wherein said time varying element changes said predetermined interval of time without changing said specified amount.

8. The system as in claim **1**, wherein said system clock includes a main part and a fractional part, including lower resolution information than is contained in said main part, and wherein said main part is used to set said times.

9. The system as in claim **8**, wherein said time varying element modifies said fractional part.

10. The system of claim **1**, wherein said commanding element comprises a manual input device.

11. The system of claim **1**, wherein said commanding element comprises an automated system which synchronizes with music being played.

12. The system of claim **11**, wherein said automated system receives at least one MIDI note stream, and synchronizes with some aspect of said at least one MIDI note stream.

13. The system of claim **12**, comprising determining a MIDI event, determining whether said MIDI event fits with a current tempo by a predetermined amount, and if not, adjusting said tempo to synchronize with the MIDI event.

14. A distributed processing varying system, comprising:

first and second distributed processing objects, each including lighting devices which are remotely controllable, said first and second distributed objects each including a processor which is commanded at certain times to carry out certain operations;

a lighting controller, connected to said first and second distributed processing objects, said controller operating

to provide commands to carry out said operations to said first and second distributed objects, said controller comprising a controlling system which commands operations to be carried out at determined times, a time changing element, receiving an indication of a new time synchronizing indicia, and a time synchronizing device, responsive to said time changing element, for synchronizing said operations with the predetermined times, said time synchronizing unit being changed by said time changing unit to thereby change the times when operations occur in both said first and second distributed processing units.

15. The system as in claim 14, further comprising a system clock, maintaining a global time, wherein said time changing element includes a first routine that increments said clock each time run, and a second routine that varies times when said first routine is run.

16. The system as in claim 14, further comprising a system clock, maintaining a global time as a highly resolved part and a lower resolved part, wherein said time changing element changes the lower resolved part.

17. The system as in claim 16, wherein said higher resolved part is an integer part used for determining timing, and said lower resolved part is a fractional part.

18. The system as in claim 14, further comprising an element which determines synchronization with an accompanying musical program, and provides said information to said time changing element.

19. A method of changing a timing of a system, comprising:

synchronizing a plurality of lighting events occuring at different separated locations with times that are set in a system clock;

determining if said system clock needs to be modified to cause said lighting events to occur at a desired tempo; and

modifying said system clock if modification is necessary.

20. A method as in claim 19, wherein said modifying comprises changing an amount of time between updates of a system clock without changing an amount that the system clock is updated at each said update.

21. A method as in claim 19, wherein said modifying comprises changing a fractional amount of clock update.

22. A method as in claim 19, wherein said determining comprises a manual action of changing a setting on a manual device.

23. A method as in claim 19, wherein said determining comprises automatically synchronzing said system clock with a musical program.

24. A method as in claim 23, wherein said determining comprises synchronizing with a MIDI stream.

25. A distributed lighting system, comprising:

a plurality of computer controlled lights, each being remotely controllable;

a controller, having controls which control said plurality of computer controlled lights, and having an input representing information in MIDI format from at least one musical instrument, said controller including an element which synchronizes an internal timer with a temp of music represented by said information, and automatically changes controlling said lights according to said internal timer.

* * * * *

EXHIBIT H

US006062706A

United States Patent [19]

Owen

[11] Patent Number: 6,062,706
[45] Date of Patent: May 16, 2000

[54] **VARIABLE COLOR FLUORESCENT LIGHTING**

[75] Inventor: **Keith Owen**, Birmingham, United Kingdom

[73] Assignee: **Light & Sound Design, Ltd.**, United Kingdom

[21] Appl. No.: **09/041,100**

[22] Filed: **Mar. 9, 1998**

[51] **Int. Cl.**[7] **F21V 9/00**; F21V 23/00

[52] **U.S. Cl.** **362/231**; 362/235; 362/295

[58] **Field of Search** 362/11, 217, 223, 362/224, 231, 235, 255, 260, 295

[56] **References Cited**

U.S. PATENT DOCUMENTS

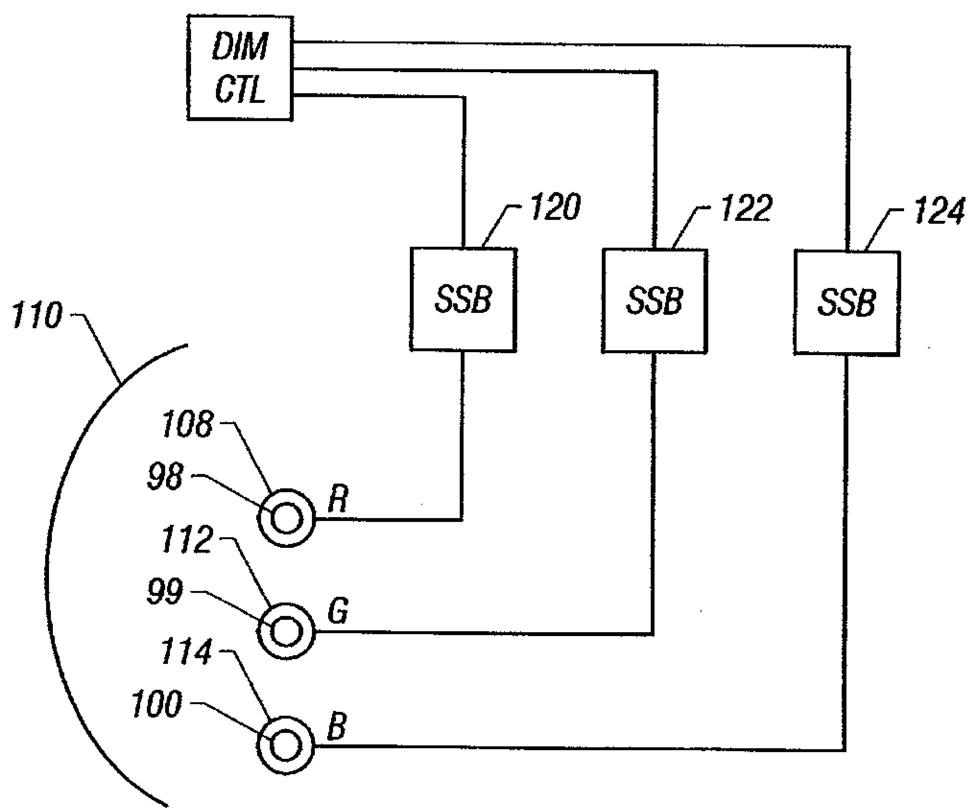
2,337,833	12/1943	Pahl	362/231
3,733,480	5/1973	Glenn	362/224
4,692,795	9/1987	Nakao	362/231
5,060,118	10/1991	Penrod et al.	362/295
5,235,497	8/1993	Costa	362/224
5,722,755	3/1998	Slape	362/231

Primary Examiner—Sandra O'Shea
Assistant Examiner—Peggy G. Neils
Attorney, Agent, or Firm—Fish & Richardson P.C.

[57] **ABSTRACT**

Three primary colored flourescent bulbs are selectively dimmed according to a ratio which produces a desired color, and at a brightness as desired.

10 Claims, 1 Drawing Sheet



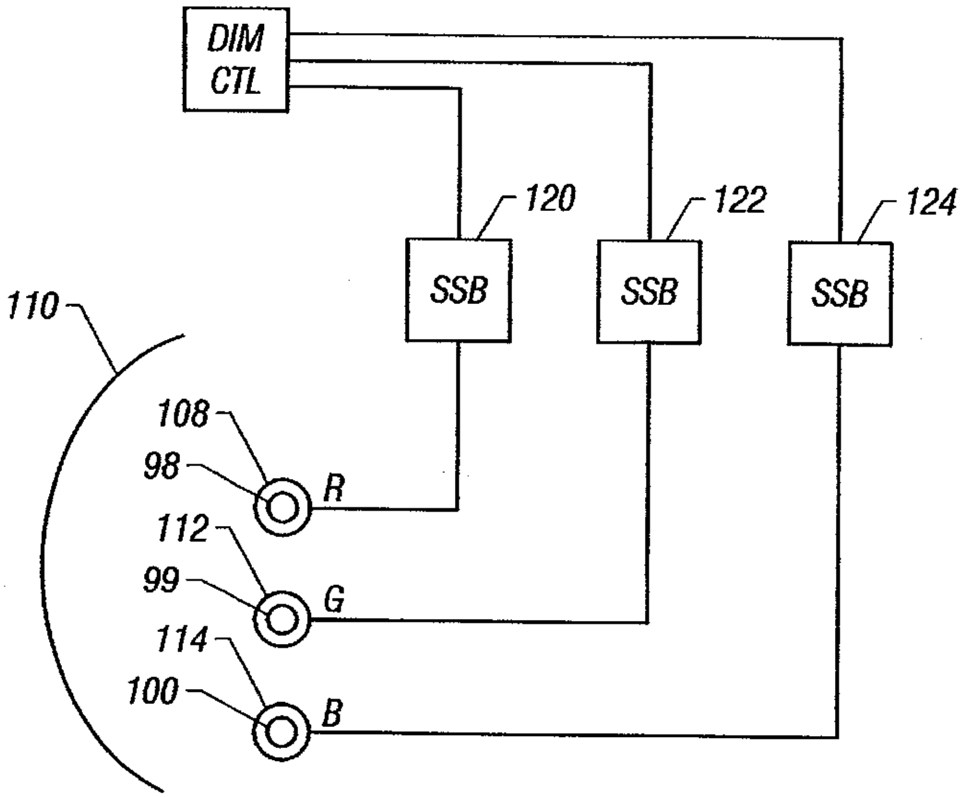

DIM
CTL
120
122
124
SSB
SSB
SSB
110
108
98
R
112
99
G
114
100
B

FIG. 1

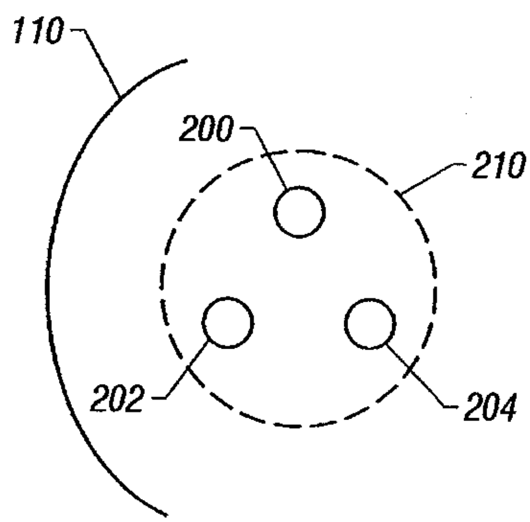

110
200
210
202
204

FIG. 2

VARIABLE COLOR FLUORESCENT LIGHTING

BACKGROUND

It is often desirable in stage lighting applications to provide a backlighting backdrop for the scene also called a "cyclorama" or simply "cyc". Such a lighting backdrop is often effected by incandescent lights which are desirably colored. The change of color allows, for example, any desired white balance for the background.

A common color changing medium used for such backlighting is a graduated gel media. The gel is rolled between two spools, and the position of unrolling of the gel determines its color. Such a device is availible from LIGHT AND SOUND DESIGN (™), Birmingham England, under the trademark of WASHLIGHT(™).

Fluorescent light is ideal for large area lighting for many reasons. First of all, fluorescent produce their lights along a relative long line. Fluorescent uniformly produces its lighting effect along that long line. This compares to many other lights which are essentially point sources. However, the fluorescent lights often dim poorly.

SUMMARY

Solid-state ballasts for fluorescent lamps are now available which allow fluorescent lamps to be dimmed between ten percent and ninety percent of their full output brightness. In addition, such solid-state ballasts often more efficiently drive the fluorescent lamp, to minimize any high frequency flickering effect.

This invention describes using primary-colored flourscent bulbs, and selectively dimming those bulbs to form a desired color effect.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 shows a first embodiment which places encapsulite-covered flourescent bulbs in a straight line with a single reflector; and

FIG. 2 shows a second embodiment with colored flourescent bulbs in triangular pattern.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

The embodiment of this invention forms a variable color fluorescent backlighting device. The embodiment is shown in FIG. 1. Three fluorescent tubes **98**, **99** and **100**, preferably cylindrical four foot or **8** foot flourescent bulbs, are mounted in appropriate fluorescent sockets within a fluorescent reflector **110**. The fluorescent tubes and reflectors are of a conventional type. The fluorescent tube **98** is driven by solid-state ballast **120**, fluorescent tube **99** is driven by solid-state ballast **122** and fluorescent tube **100** is driven by solid-state ballast **124**. Of course, all the operations could all be done using a single solid-state ballast so long as that ballast allows for separate control of dimming of each of the fluorescent lamps.

Each of the lamps produces a different primary color. This can be done by using different colored lamps. The illustrated embodiment covers each lamp with an encapsulite shield **108**, **112**, **114**. The encapsulite shield is a polycarbonate sleeve. The encapsulite sleeves are available in various colors. The fluorescent lamp **98** is covered by red encapsulite sleeve **108**. The lamp **99** is covered by green encapsulite sleeve **112**, and the lamp **98** is covered by blue encapsulite

sleeve **114**. Each of these encapsulite sleeves performs a filtering function, allowing predominantly the specific color wavelengths to be projected. This produces an effect simulating red, green and blue light. Each of the lamps is also independently controllable via the respective solid-state ballast. This allows mixing the primary colors in any desired way.

The three lights are clustered in proximity in order to provide additive mixing of the colors from the three lights. If the lights were not closely located relative to each other, the effect could produce shadows of different colors. Therefore, it is preferable for the lights to be provided sufficiently close to one another to avoid different colored shadows being produced. Preferably, the devices are co-located as a semi-point source, located with the point at the focal point of the reflector **110**.

The second embodiment shown in FIG. 2 places the three is bulbs in a substantially triangular pattern to better simulate the point source effect. The three bulbs **200**, **202**, and **204** in the FIG. 2 embodiment are respectively of different colors. All are generally located within the focal point **210** of the reflector **110**.

The resultant device, therefore, includes a red fluorescent tube, a green fluorescent tube, and a blue fluorescent tube. The colors from the tubes, in operation, will mix according to their brightnesses to form a desired color lighting effect. Any desired color can be obtained by combinations according to the so-called CIE chromaticity chart. For example, a pink color can be formed by a combination of blue, red, and green. The relative brightnesses of the colored lights are adjusted to provide 100% red, 50% blue, and just a little (25%) green. The absolute brightness of the pink color can be adjusted by adjusting the absolute brightness of the lamps while maintaining the ratios between the colors. For example, therefore, a dimmer effect with pink light could use 10% green, 20% blue, and 40% red.

By control of these two parameters: the relative brightness of the primary colors and the absolute brightness of the primary colors, any desired color and brightness within the available brightness output of the fluorescent tubes can be obtained.

Since this effect is produced by a fluorescent system, a number of advantages are obtained, including increased bulb life, color spread over a longer length, and lower cost. The light also produces a different effect than is obtained from incandescent light.

The present invention contemplates use with either standard fluorescent fixtures, or high output ("HO") or very high output ("VHO") fluorescent lamps to produce greater light amounts.

Other embodiments are within the disclosed elements. For instance, although only a few different color combinations are disclosed, it should be apparent that anyone of ordinary skill in the art of color mixing would understand how to form any desired color from red, green and blue combinations. The present invention describes use with a solid-state ballast, but it should be understood that any ballast which allows relative dimming of the lamps could alternately be used. Other primary colors, e.g., cyan, magenta, and yellow could alternately be used.

What is claimed is:

1. A fluorescent backlighting system, comprising:

a light reflector having a curved reflective surface that has a focus position;

a plurality of fluorescent lighting devices disposed in close proximity with each other near said focus position

of said curved reflective surface, said fluorescent lighting devices including at least a first fluorescent lighting device which produces a first primary color and a second fluorescent lighting device which produces a second primary color;

a solid state fluorescent ballast, controlling relative brightness outputs of said first and second fluorescent devices, and dimmable to change an output of light output of said first and second fluorescent lighting devices; and

a dim control device, connected to said solid state ballast and controlling said outputs of said solid state ballast to allow turning on said first fluorescent device at a higher brightness output than said second fluorescent device to thereby enable different colors to be produced.

2. A system as in claim 1, wherein said fluorescent lighting devices are cylindrical bulbs.

3. A system as in claim 1, further comprising a third fluorescent lighting device, having a different color output than said first and second fluorescent lighting devices, and where said ballast and said dim control device each also control said third flourescent lighting device.

4. A system as in claim 3, wherein said first, second and third flourescent lighting devices are respectively primary colors.

5. A system as in claim 3, wherein said first, second and third flourescent lighting devices are red, green and blue.

6. A flourescent lighting system, comprising:

first, second and third flourescent bulbs, each having a different color light output, and located in an area;

a light reflector having a single reflective surface and positioned relative to said fluorescent bulbs to have a focus position of said reflector in said area, collectively reflecting said light from said area to produce a light output which sums light output from said first second and third bulbs;

first, second and third dimmable flourescent ballast parts, each respectively driving one of said flourescent bulbs, and each separately controllable to change an amount of light output from each of said bulbs, to allow any color, and any intensity output, to be produced.

7. A method of flourscent lighting, comprising:

obtaining first, second and third flourescent bulbs, respectively having first, second and third colors, and a light reflector having a focus position, collectively reflecting said light from said bulbs to produce a light output which sums light output from said first second and third bulbs;

arranging said first, second, and third flourescent bulbs in a close proximity of said focus position;

determining a desired light color to be output, and determining a ratio between said first, second and third colors to produce said desired light color;

determining a desired light intensity to be output;

using three dimmable flourescent ballasts to drive said first, second and third bulbs at said ratio and at said desired light intensity.

8. A system as in claim 3, wherein said first, second, and third fluorescent lighting devices are arranged in a triangular geometry.

9. A system as in claim 6, wherein said first, second, and third fluorescent bulbs are arranged in said area to form a triangular geometry.

10. A method as in claim 7, further comprising arranging said first, second, and third fluorescent bulbs in said area to form a triangular geometry.

* * * * *

EXHIBIT I

US006256136B1

(12) **United States Patent**
Hunt

(10) **Patent No.: US 6,256,136 B1**
(45) **Date of Patent: Jul. 3, 2001**

(54) **PIXEL BASED GOBO RECORD CONTROL FORMAT**

(75) Inventor: **Mark Hunt**, Derby (GB)

(73) Assignee: **Light & Sound Design, Ltd.**, Birmingham (GB)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/500,393**

(22) Filed: **Feb. 8, 2000**

Related U.S. Application Data

(63) Continuation of application No. 09/145,314, filed on Aug. 31, 1998, now Pat. No. 6,057,958.

(60) Provisional application No. 60/065,133, filed on Nov. 12, 1997, and provisional application No. 60/059,161, filed on Sep. 17, 1997.

(51) **Int. Cl.**[7] **G02B 26/00**

(52) **U.S. Cl.** **359/291**; 345/418; 345/431; 382/162; 382/181; 348/26; 348/110; 364/148; 364/188

(58) **Field of Search** 359/291; 345/418, 345/430, 431; 382/181, 162, 190; 348/110, 26, 649; 364/148, 188

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,828,485		10/1998	Hewlett	359/291
5,852,443	*	12/1998	Kenworthy	345/431

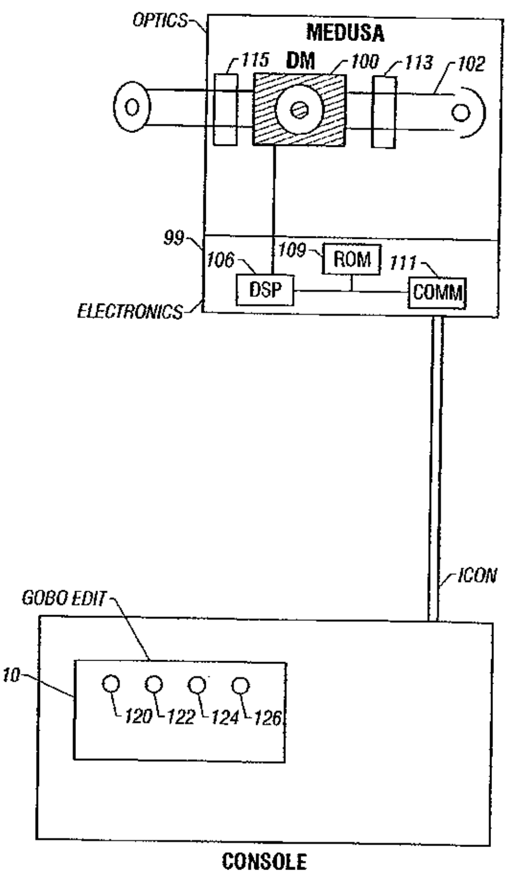
* cited by examiner

Primary Examiner—Loha Ben
(74) *Attorney, Agent, or Firm*—Fish & Richardson P.C.

(57) **ABSTRACT**

A special record format used for commanding light pattern shapes and addressable light pattern shape generator. The command format includes a first part which commands a specified gobo and second parts which command the characteristics of that gobo. The gobo is formed by making a default gobo based on the type and modifying that default gobo to fit the characteristics.

111 Claims, 6 Drawing Sheets



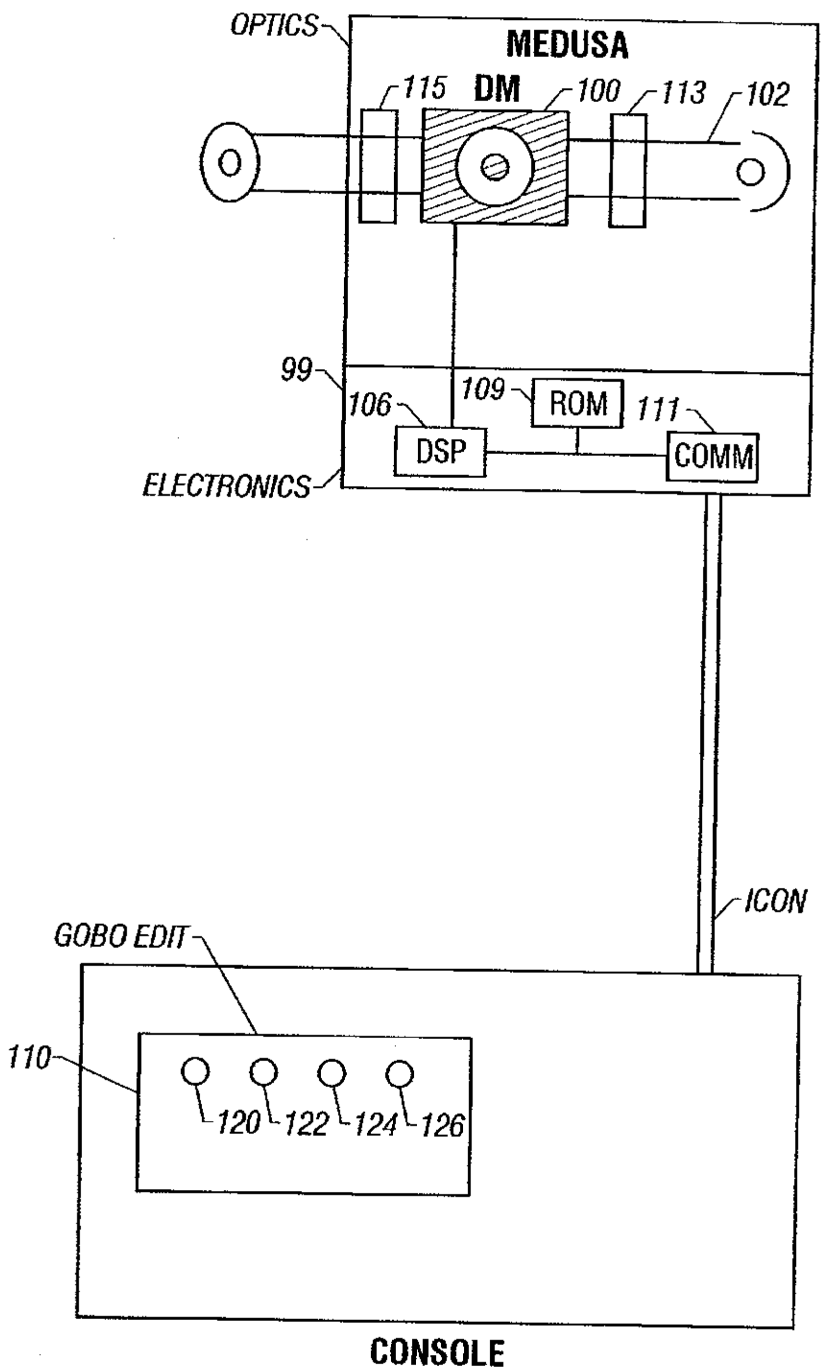

OPTICS
MEDUSA
115
DM
100
113
102
99
106
109
ROM
111
DSP
COMM
ELECTRONICS
ICON
GOBO EDIT
110
120
122
124
126
CONSOLE

FIG. 1

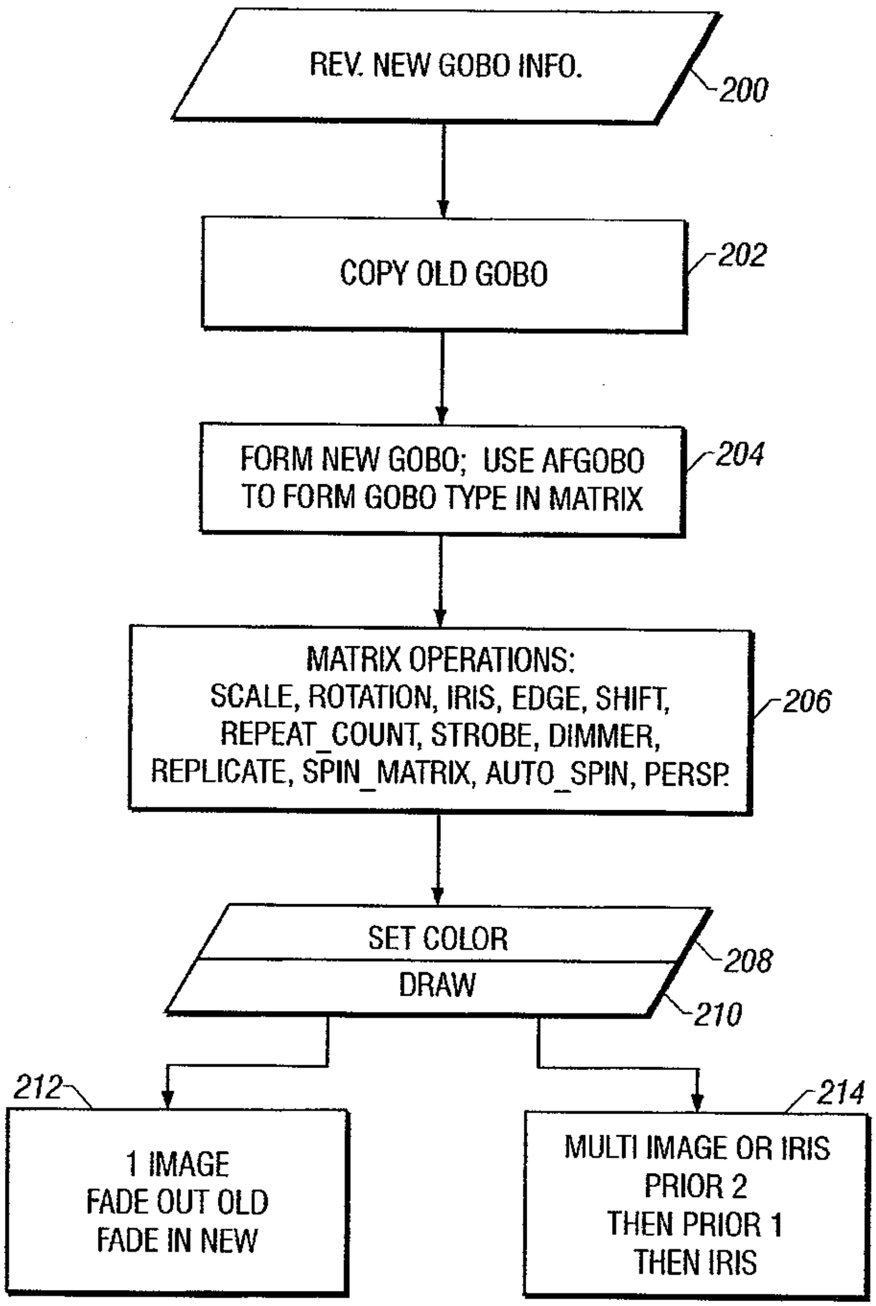

REV. NEW GOBO INFO.
200
COPY OLD GOBO
202
FORM NEW GOBO; USE AFGOBO TO FORM GOBO TYPE IN MATRIX
204
MATRIX OPERATIONS:
SCALE, ROTATION, IRIS, EDGE, SHIFT, REPEAT_COUNT, STROBE, DIMMER, REPLICATE, SPIN_MATRIX, AUTO_SPIN, PERSP.
206
SET COLOR
208
DRAW
210
212
1 IMAGE
FADE OUT OLD
FADE IN NEW
214
MULTI IMAGE OR IRIS
PRIOR 2
THEN PRIOR 1
THEN IRIS

FIG. 2

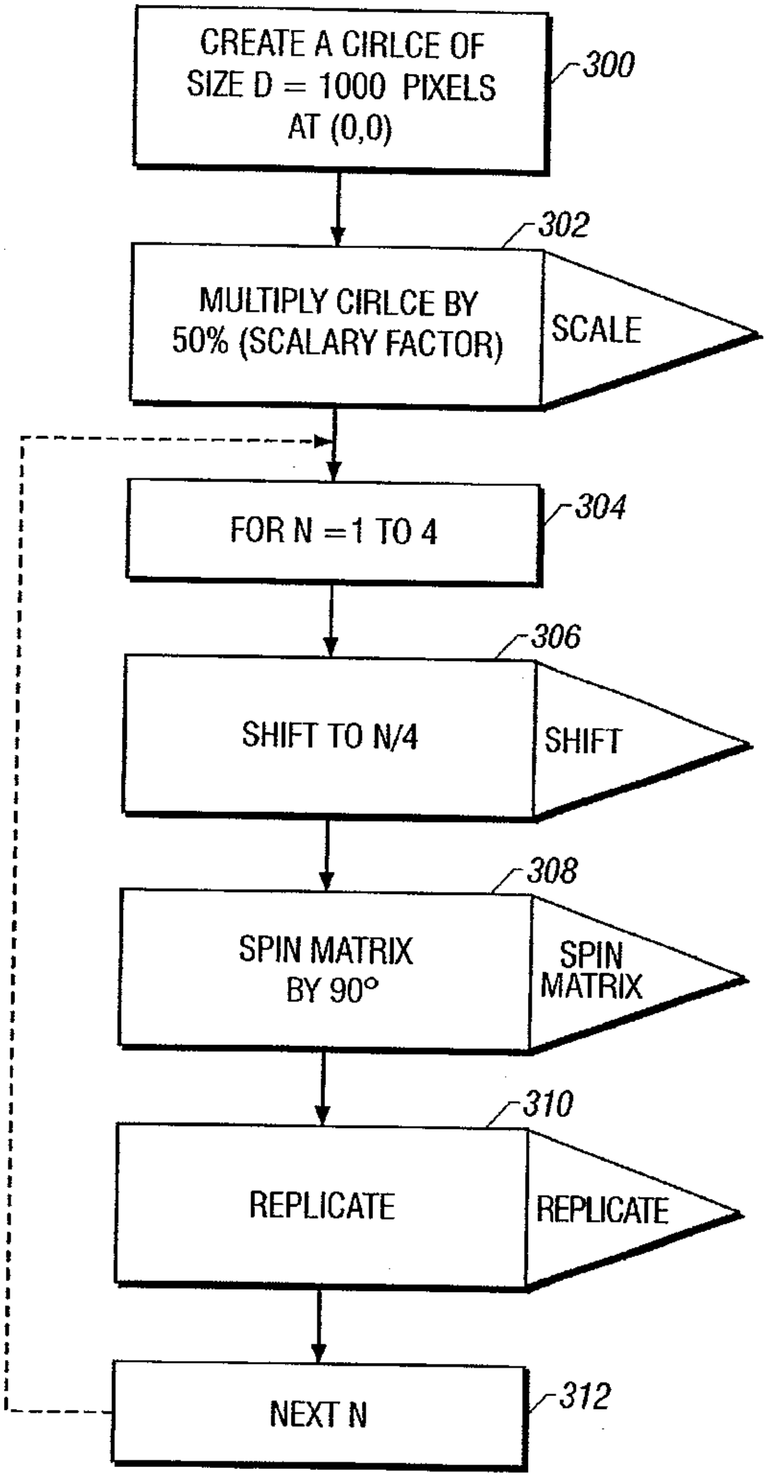

CREATE A CIRLCE OF SIZE D = 1000 PIXELS AT (0,0)
300
302
MULTIPLY CIRLCE BY 50% (SCALARY FACTOR)
SCALE
FOR N =1 TO 4
304
306
SHIFT TO N/4
SHIFT
308
SPIN MATRIX BY 90°
SPIN MATRIX
310
REPLICATE
REPLICATE
NEXT N
312

FIG. 3

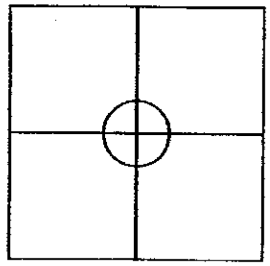


FIG. 4A

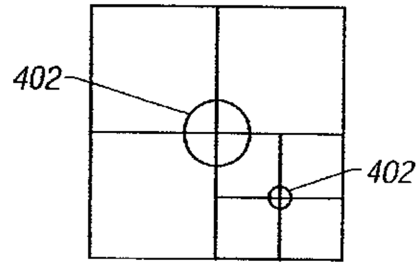


FIG. 4E

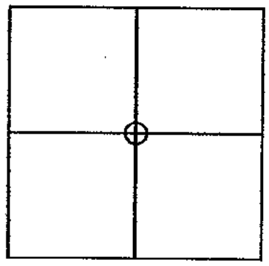


FIG. 4B

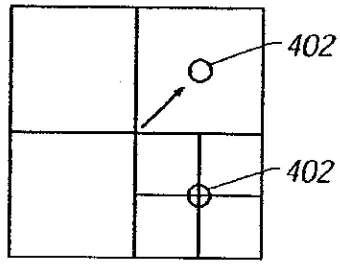


FIG. 4F

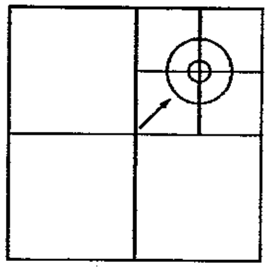


FIG. 4C

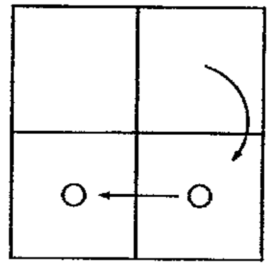


FIG. 4G

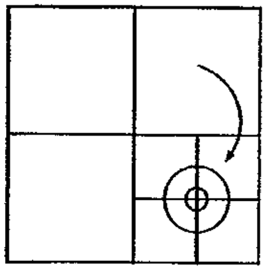


FIG. 4D

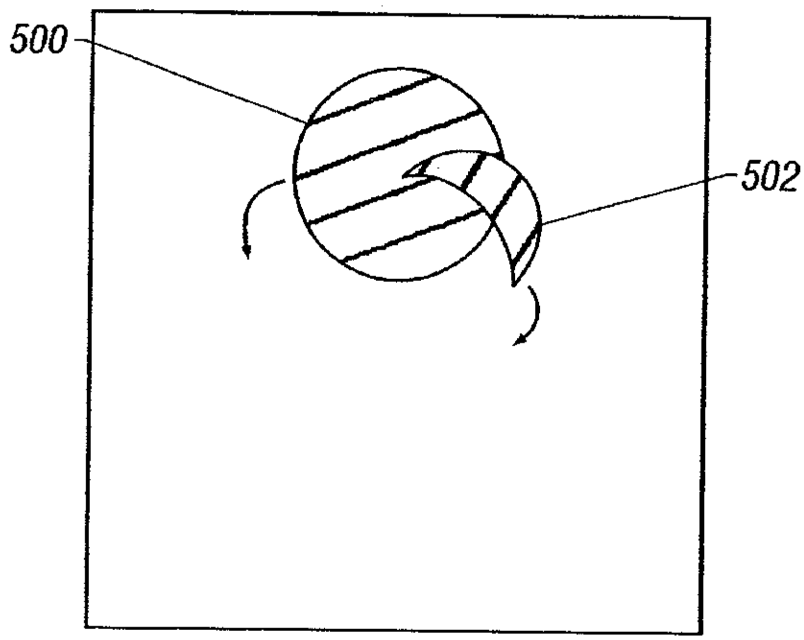
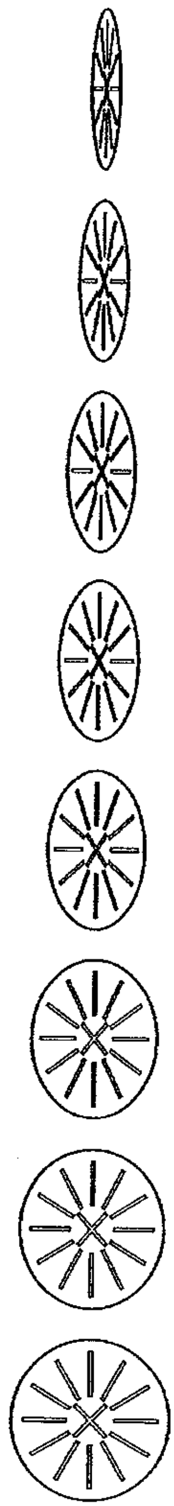


FIG. 5



1 2 3 4 5 6 7 8

FIG. 6

PIXEL BASED GOBO RECORD CONTROL FORMAT

This is a continuation of U.S. application Ser. No. 09/145,314, filed Aug. 31, 1998, now U.S. Pat. No. 6,057, 958, which claims priority from provisional application No. 60/059,161, filed Sep. 17, 1997; and, provisional application No. 60/065,133, filed Nov. 12, 1997.

FIELD

The present invention relates to a system of controlling light beam pattern ("gobo") shape in a pixilated gob0 control system.

BACKGROUND

Commonly assigned patent application Serial No. 08/854, 353, now U.S. Pat. No. 6,188,933, the disclosure of which is herewith incorporated by reference, describes a stage lighting system which operates based on computer-provided commands to form special effects. One of those effects is control of the shape of a light pattern that is transmitted by the device. This control is carried out on a pixel-by-pixel basis, hence referred to in this specification as pixilated. Control is also carried out using an x-y controllable device. The preferred embodiment describes using a digital mirror device, but other x-y controllable devices such as a grating light valve, are also contemplated.

The computer controlled system includes a digital signal processor 106 which is used to create an image command. That image command controls the pixels of the x-y controllable device to shape the light that it is output from the device.

The system described in the above-referenced application allows unparalleled flexibility in selection of gobo shapes and movement. This opens an entirely new science of controlling gobos. The present inventors found that, unexpectedly, even more flexibility is obtained by a special control language for controlling those movements.

SUMMARY

The present disclosure defines a way of communicating with an x-y controllable device to form special electronic light pattern shapes. More specifically, the present application describes using a control language to communicate with an electronic gobo in order to reposition part or all of the image that is shaping the light.

BRIEF DESCRIPTION OF THE DRAWINGS

These and other aspects of the invention will now be described with reference to the attached drawings, in which:

FIG. 1 shows a block diagram of the basic system operating the embodiment;

FIG. 2 shows a basic flowchart of operation;

FIG. 3 shows a flowchart of forming a replicating circles type gobo;

FIGS. 4A through 4G show respective interim results of carrying out the replicating circles operation;

FIG. 5 shows the result of two overlapping gobos rotating in opposite directions; and

FIGS. 6(1) through 6(8) show a z-axis flipping gobo.

DESCRIPTION OF THE PREFERRED EMBODIMENT

FIG. 1 shows a block diagram of the hardware used according to the preferred embodiment. As described above,

this system uses a digital mirror device 100, which has also been called a digital mirror device ("DMD") and a digital light processor device ("DLP"). More generally, any system which allows controlling shape of light on a pixel basis, including a grating light valve, could be used as the light shaper. This light shaper forms the shape of light which is transmitted. FIG. 1 shows the light being transmitted as 102, and shows the transmitted light. The information for the digital mirror 100 is calculated by a digital signal processor 106. Information is calculated based on local information stored in the lamp, e.g., in ROM 109, and also in information which is received from the console 104 over the communication link.

The operation is commanded according to a format.

The preferred data format provides 4 bytes for each of color and gobo control information.

The most significant byte of gobo control data, ("dfGobo") indicates the gobo type. Many different gobo types are possible. Once a type is defined, the gobo formed from that type is represented by a number. That type can be edited using a special gobo editor described herein. The gobo editor allows the information to be modified in new ways, and forms new kinds of images and effects.

The images which are used to form the gobos may have variable and/or moving parts. The operator can control certain aspects of these parts from the console via the gobo control information. The type of gobo controls the gobo editor to allow certain parameters to be edited.

The examples given below are only exemplary of the types of gobo shapes that can be controlled, and the controls that are possible when using those gobo shapes. of course, other controls of other shapes are possible and predictable based on this disclosure.

A first embodiment is the control of an annulus, or "ring" gobo. The DMD 100 in FIG. 1 is shown with the ring gobo being formed on the DMD. The ring gobo is type 000A. When the gobo type 0A is enabled, the gobo editor 110 on the console 104 is enabled and the existing gobo encoders 120, 122, 124, and 126 are used. The gobo editor 110 provides the operator with specialized control over the internal and the external diameters of the annulus, using separate controls in the gobo editor.

The gobo editor and control system also provides other capabilities, including the capability of timed moves between different edited parameters. For example, the ring forming the gobo could be controlled to be thicker. The operation could then effect a timed move between these "preset" ring thicknesses. Control like this cannot even be attempted with conventional fixtures.

Another embodiment is a composite gobo with moving parts. These parts can move though any path that are programmed in the gobo data itself. This is done in response to the variant fields in the gobo control record, again with timing. Multiple parts can be linked to a single control allowing almost unlimited effects.

Another embodiment of this system adapts the effect for an "eye" gobo, where the pupil of the eye changes its position (look left, look right) in response to the control.

Yet another example is a Polygon record which can be used for forming a triangle or some other polygonal shape.

The control can be likened to the slider control under a QuickTime movie window, which allows you to manually move to any point in the movie. However, our controls need not be restricted to timelines.

Even though such moving parts are used, scaling and rotation on the gobo is also possible.

The following type assignments are contemplated:
00-0F=FixedGobo (with no "moving parts")
10-1F=SingleCntrl (with 1"moving part")
20-2F=DoubleCntrl (with 2"moving parts")
30-FF=undefined, reserved.

The remaining control record bytes for each type are defined as follows:

Byte:	dfGobo2	dfGobo3	dfGobo4	#gobos/type,	total memory
FixedGobo:	ID[23:16]	ID[15:8]	ID[7:0]	16M/type	256M
SingleCntrl:	ID[15:8]	ID[7:0]	control #1	64k/type	1M
DoubleCntrl:	ID[7:0]	control #2	control #1	256/type	4k

As can be seen from this example, this use of the control record to carry control values does restrict the number of gobos which can be defined of that type, especially for the 2-control type.

Console Support:

The use of variant part gobos requires no modifications to existing console software for the ICON (7M) console. The Gobo editor in current ICON software already provides 4 separate encoders for each gobo. These translate directly to the values of the 4 bytes sent in the communications data packet as follows:

Byte:	dfGobo	dfGobo2	dfGobo3	dfGobo4
Enc:	TopRight	MidRight	BotRight	BotLeft
FixedGobo:		ID[23:16]	ID[15:8]	ID[7:0]
SingleCntrl:		ID[15:8]	ID[7 0]	control #1
DoubleCntrl:		ID[7:0]	control #2	control #1

These values would be part of a preset gobo, which could be copied as the starting point.

Once these values are set, the third and fourth channels automatically become the inner/outer radius controls. Using two radii allows the annulus to be turned "inside out".

Each control channel's data always has the same meaning within the console. The console treats these values as simply numbers that are passed on. The meanings of those numbers, as interpreted by the lamps change according to the value in dfGobo.

The lamp will always receives all 4 bytes of the gobo data in the same packet. Therefore, a "DoubleCntrl" gobo will always have the correct control values packed along with it.

Hence, the console needs no real modification. If a "soft" console is used, then name reassignments and/or key reassignments may be desirable.

Timing:

For each data packet, there is an associated "Time" for gobo response. This is conventionally taken as the time allotted to place the new gobo in the light gate. This delay has been caused by motor timing. In this system, variant gobo, the control is more dynamically used. If the non-variant parts of the gobo remain the same, then it is still the same gobo, only with control changes. Then, the time value is interpreted as the time allowed for the control change.

Since different gobo presets (in the console) can reference the same gobo, but with different control settings, this allows easily programmed timed moves between different annuli, etc. Internal Workings:

When the gobo command data is extracted from the packet at the lamp, the dfGobo byte is inspected first, to see if either dfGobo3 or dfGobo4 are significant in selecting the

image. In the case of the "Cntrl" variants, one or both of these bytes is masked out, and the resulting 32-bit number is used to search for a matching gobo image (by Gobo _1D) in the library stored in the lamp's ROM **109**.

If a matching image is found, and the image is not already in use, then the following steps are taken:

1) The image data is copied into RAM, so that its fields may be modified by the control values. This step will be skipped if the image is currently active.
2) The initial control values are then recovered from the data packet, and used to modify certain fields of the image data, according to the control records.
3) The image is drawn on the display device, using the newly-modified fields in the image data.

If the image is already in use, then the RAM copy is not altered. Instead, a time-sliced task is set up to slew from the existing control values to those in the new data packet, in a time determined by the new data packet. At each vertical retrace of the display, new control values are computed, and steps 2 (using the new control values) and 3 above are repeated, so that the image appears modified with time.

The Image Data Records:

All images stored in the lamp are in a variant record format:

Header:
Length 32 bits, offset to next gobo in list.
Gobo _1D 32 bits, serial number of gobo.
Gobo Records:
Length 32 bits, offset to next record.
Opcode 16 bits, type of object to be drawn.
Data Variant part—data describing object.
_Length 32 bits, offset to next record.
Opcode 16 bits, type of object to be drawn.
Data Variant part—data describing object.
_EndMarker 64 bits, all zeroes—indicates end of gobo data.
+Next gobo, or End Marker, indicating end of gobo list.

Gobos with controls are exactly the same, except that they contain control records, which describe how the control values are to affect the gobo data. Each control record contains the usual length and Opcode fields, and a field containing the control number (1 or 2).

These are followed by a list of "field modification" records. Each record contains information about the offset (from the start of the gobo data) of the field, the size (8, 16 or 32 bits) of the field, and how its value depends on the control value.

Length 32 bits, offset to next record
Opcode 16 bits=control_record (constant)
CntrlNum 16 bits=1 or 2 (control number)
/*field modification record #1 */
Address 16 bits, offset from start of gobo to affected field.
Flags 16 bits, information about field (size, signed, etc)
Scale 16 bits, scale factor applied to control before use
zPoint 16 bits, added to control value after scaling.
/* field modification record #2 */
Address 16 bits, offset from start of gobo to affected field.
Flags 16 bits, information about field (size, signed, etc)
Scale 16 bits, scale factor applied to control before use
zPoint 16 bits, added to control value after scaling.

As can be seen, a single control can have almost unlimited effects on the gobo, since ANY values in the data can be modified in any way, and the number of field modification records is almost unlimited.

Note that since the control records are part of the gobo data itself, they can have intimate knowledge of the gobo structure. This makes the hard-coding of field offsets acceptable.

In cases where the power offered by this simple structure is not sufficient, a control record could be defined which contains code to be executed by the processor. This code would be passed parameters, such as the address of the gobo data, and the value of the control being adjusted.

Example Records.

The Annulus record has the following format:

Length 32 bits
Opcode 16 bits, =type_annulus
Pad 16 bits, unused
Centre_x 16 bits, x coordinate of centre
Centre_y 16 bits, y coordinate of centre
OuterRad 16 bits, outside radius (the radii get swapped when drawn if their values are in the wrong order)
InnerRad 16 bits, inside radius

It can be seen from this that it is easy to "target" one of the radius parameters from a control record. Use of two control records, each with one of the radii as a target, would provide full control over the annulus shape.

Note that if the centre point coordinates are modified, the annulus will move around the display area, independent of any other drawing elements in the same gobo's data.

The Polygon record for a triangle has this format:

Length 32 bits
Opcode 16 bits, =type_polygon
Pad 16 bits, vertex count=3
Centre_x 16 bits, x coordinate of vertex
Centre_y 16 bits, y coordinate of vertex
Centre_x 16 bits, x coordinate of vertex
Centre_y 16 bits, y coordinate of vertex
Centre_x 16 bits, x coordinate of vertex
Centre_y 16 bits, y coordinate of vertex

It is easy to modify any of the vertex coordinates, producing distortion of the triangle.

The gobo data can contain commands to modify the drawing environment, by rotation, scaling, offset, and color control, the power of the control records is limitless.

Second Embodiment

This second embodiment provides further detail about implementation once the gobo information is received.

Gobo information is, at times, being continuously calculated by DSP **106**. The flowchart of FIG. **2** shows the handling operation that is carried out when new gobo information is received.

At step **200**, the system receives new gobo information. In the preferred embodiment, this is done by using a communications device 111 in the lamp **99**. The communications device is a mailbox which indicates when new mail is received. Hence, the new gobo information is received at step **200** by determining that new mail has been received.

At step **202**, the system copies the old gobo and switches pointers. The operation continues using the old gobo until the draw routine is called later on.

At step **204**, the new information is used to form a new gobo. The system uses a defined gobo ("dfGobo") as discussed previously which has a defined matrix. The type dfGobo is used to read the contents from the memory **109** and thereby form a default image. That default image is formed in a matrix. For example, in the case of an annulus, a default size annulus can be formed at position 0,0 in the matrix. An example of forming filled balls is provided herein.

Step **206** represents calls to subroutines. The default gobo is in the matrix, but the power of this system is its ability to very easily change the characteristics of that default gobo. In this embodiment, the characteristics are changed by changing the characteristics of the matrix and hence, shifting that default gobo in different ways. The matrix operations, which are described in further detail herein, include scaling the gobo, rotation, iris, edge, strobe, and dimmer. Other matrix operations are possible. Each of these matrix operations takes the default gobo, and does something to it.

For example, scale changes the size of the default gobo rotation rotates the default gobo by a certain amount

Iris simulates an iris operation by choosing an area of interest, typically circular, and erasing everything outside that area of interest. This is very easily done in the matrix, since it simply defines a portion in the matrix where all black is written.

Edge effects carry out certain effects on the edge such as softening the edge. This determines a predetermined thickness, which is translated to a predetermined number of pixels, and carries out a predetermined operation on the number of pixels. For example, for a 50% edge softening, every other pixel can be turned off. The strobe is in effect that allows all pixels to be turned on and off at a predetermined frequency, i.e., 3 to 10 times a second. The dimmer allows the image to be made dimmer by turning off some of the pixels at predetermined times.

The replicate command forms another default gobo, to allow two different gobos to be handled by the same record. This will be shown with reference to the exemplary third embodiment showing balls. Each of those gobos is then handled as the same unit and the entirety of the gobos can be, for example, rotated. The result of step **206** and all of these subroutines that are called is that the matrix includes information about the bits to be mapped to the digital mirror **100**.

At step **208**, the system then obtains the color of the gobos from the control record discussed previously. This gobo color is used to set the appropriate color changing circuitry **113** and **115** in the lamp **99**. Note that the color changing circuitry is shown both before and after the digital mirror **100**. It should be understood that either of those color changing circuits could be used by itself.

At step **210**, the system calls the draw routine in which the matrix is mapped to the digital mirror. This is done in different ways depending on the number of images being used. Step **212** shows the draw routine for a single image being used as the gobo. In that case, the old gobo, now copied as shown in step **202**, is faded out while the new gobo newly calculated is faded in. Pointers are again changed so that the system points to the new gobo. Hence, this has the effect of automatically fading out the old gobo and fading in the new gobo.

Step **214** schematically shows the draw routine for a system with multiple images for an iris. In that system, one of the gobos is given priority over the other. If one is brighter than the other, then that one is automatically given priority. The one with priority 2, the lower priority one, is written first. Then the higher priority gobo is written. Finally, the iris is written which is essentially drawing black around the edges of the screen defined by the iris. Note that unlike a conventional iris, this iris can take on many different shapes. The iris can take on not just a circular shape, but also an elliptical shape, a rectangular shape, or a polygonal shape. In addition, the iris can rotate when it is non-circular so that for the example of a square iris, the edges of the square can actually rotate.

Returning to step **206**, in the case of a replicate, there are multiple gobos in the matrix. This allows the option of spinning the entire matrix, shown as spin matrix.

An example will now be described with reference to the case of repeating circles. At step **200**, the new gobo infor-

mation is received indicating a circle. This is followed by the other steps of **202** where the old gobo is copied, and **204** where the new gobo is formed. The specific operation forms a new gobo at step **300** by creating a circle of size diameter equals 1000 pixels at origin 00. This default circle is automatically created. FIG. **4A** shows the default gobo which is created, a default size circle at 00. It is assumed for purposes of this operation that all of the circles will be the same size.

At step **302**, the circle is scaled by multiplying the entire circle by an appropriate scaling factor. Here, for simplicity, we are assuming a scaling factor of 50% to create a smaller circle. The result is shown in FIG. 4B. A gobo half the size of the gobo of FIG. **4A** is still at the origin. This is actually the scale of the subroutine as shown in the right portion of step **302**. Next, since there will be four repeated gobos in this example, a four-loop is formed to form each of the gobos at step **304**. Each of the gobos is shifted in position by calling the matrix operator shift. In this example, the gobo is shifted to a quadrant to the upper right of the origin. This position is referred to as π over 4 in the FIG. **3** flowchart and results in the gobo being shifted to the center portion of the top right quadrant as shown in FIG. 4C. This is again easily accomplished within the matrix by moving the appropriate values. At step **308**, the matrix is spun by 90 degrees in order to put the gobo in the next quadrant as shown in FIG. **4D** in preparation for the new gobo being formed into the same quadrant. Now the system is ready for the next gobc, thereby calling the replicate command which quite easily creates another default gobo circle and scales it. The four-loop is then continued at step **312**.

The replicate process is shown in FIG. **4E** where a new gobo **402** is formed in addition to the existing gobo **400**. The system then passes again through the four-loop, with the results being shown in the following figures. In FIG. **4F**, the new gobo **402** is again moved to the upper right quadrant (step **306**). In FIG. 4G, the matrix is again rotated to leave room for a new gobo in the upper right quadrant. This continues until the end of the four-loop. Hence, this allows each of the gobos to be formed.

Since all of this is done in matrix operation, it is easily programmable into the digital signal processor. While the above has given the example of a circle, it should be understood that this scaling and moving operation can be carried out for anything. The polygons, circles, annulus, and everything else is easily scaled.

The same operation can be carried out with the multiple parameter gobos. For example, for the case of a ring, the variable takes the form annulus (inner R, outer R, x and y). This defines the annulus and turns of the inner radius, the outer radius, and x and y offsets from the origin. Again, as shown in step **3**, the annulus is first written into the matrix as a default size, and then appropriately scaled and shifted. In terms of the previously described control, the ring gobo has two controls: control **1** and control **2** defined the inner and outer radius.

Each of these operations is also automatically carried out by the command repeat count which allows easily forming the multiple position gobo of FIGS. **4A–4G**. The variable auto spin defines a continuous spin operation. The spin operation commands the digital signal processor to continuously spin the entire matrix by a certain amount each time.

One particularly interesting feature available from the digital mirror device is the ability to use multiple gobos which can operate totally separately from one another raises the ability to have different gobos spinning in different directions. When the gobos overlap, the processor can also calculate relative brightness of the two gobos. In addition, one gobo can be brighter than the other. This raises the possibility of a system such as shown in FIG. **5**. Two gobos are shown spinning in opposite directions: the circle gobo **500** is spinning in the counterclockwise direction, while the half moon gobo **502** is spinning in the clockwise direction. At the overlap, the half moon gobo which is brighter than the circle gobo, is visible over the circle gobo. Such effects were simply not possible with previous systems. Any matrix operation is possible, and only a few of those matrix operations have been described herein.

A final matrix operation to be described is the perspective transformation. This defines rotation of the gobo in the Z axis and hence allows adding depth and perspective to the gobo. For each gobo for which rotation is desired, a calculation is preferably made in advance as to what the gobo will look like during the Z axis transformation. For example, when the gobo is flipping in the Z axis, the top goes back and looks smaller while the front comes forward and looks larger. FIGS. **6(1)–6(8)** show the varying stages of the gobo flipping. In FIG. **6(8)**, the gobo has its edge toward the user. This is shown in FIG. **6(8)** as a very thin line, e.g., three pixels wide, although the gobo could be zero thickness at this point. Automatic algorithms are available for such Z axis transformation, or alternatively a specific Z axis transformation can be drawn and digitized automatically to enable a custom look.

Although only a few embodiments have been described in detail above, other embodiments are contemplated by the inventor and are intended to be encompassed within the following claims. In addition, other modifications are contemplated and are also intended to be covered.

What is claimed is:

1. A method of controlling an addressable light pattern generator, comprising:

defining a gobo type, that represents at least a pattern of a gobo represented by said gobo type, using information indicative of said type; and

changing at least one value of the gobo represented by said type and providing a different display of said gobo, having the same general pattern, but having different values.

2. A method as in claim **1**, wherein said type is represented by a number.

3. A method as in claim **1**, wherein said value represents a size of a part of the gobo.

4. A method as in claim **1**, wherein said value represents a number of parts of said gobo.

5. A method as in claim **1**, wherein said gobo type represents a general outline of said gobo.

6. A method as in claim **1**, wherein said gobo is a type which has a variable number of parts and said value represents said number of parts.

7. A method as in claim **1**, wherein said gobo is a ring-shaped gobo.

8. A method as in claim **7**, wherein said value is a size of a diameter of said gobo.

9. A method as in claim **1**, wherein said gobo is a polygon.

10. A method as in claim **9**, wherein said value represents a number of sides of said polygon.

11. A method as in claim **1**, further comprising a type assignment which indicates a number of moving parts of said gobo.

12. A method as in claim **11**, wherein said number of moving parts can be zero moving parts, one moving part, or two moving parts.

13. A method as in claim **1**, further comprising commanding rotating the gobo.

14. A method as in claim 1, further comprising commanding scaling the gobo.

15. A method as in claim 1, further comprising commanding changing a color of the gobo.

16. A console having a first control part allowing selection of a gobo type and having a second control part allowing editing of a gobo selected by said gobo type.

17. A console as in claim 16, wherein said second control part enables gobo size to be edited.

18. A console as in claim 16, wherein said gobo type is one which has multiple parameters, and said second control part allows said multiple parameters to be edited.

19. A console as in claim 18, wherein said multiple parameters include inner and outer diameters which can be separately edited.

20. A console as in claim 16, wherein said gobo includes a variable number of parts and said second control part includes an editor for s aid number of parts.

21. A console as in claim 20, wherein said number of parts includes a number of actual shapes.

22. A console as in claim 20, wherein said number of parts includes a number of sides of a polygonal shape.

23. A console as in claim 16, wherein said second control part includes encoders which have functions that depend on the gobo number.

24. A console as in claim 16, further comprising a memory, storing gobo record images.

25. A console as in claim 24, wherein said gobo record images include a code representing a type of object to be drawn.

26. A console as in claim 16, further comprising a digital signal processor calculating gobo information.

27. A method comprising:

receiving digital information indicative of a gobo to be used to shape a light beam;

copying a current gobo being used and continuing to use the current gobo to shape the light beam;

using the information to form a new gobo; and

switching from said old gobo to said new gobo to thereafter shape the light beam with said new gobo.

28. A method as in claim 27, further drawing the new gobo by forming a default image of a new gobo type.

29. A method as in claim 28, wherein said default image is formed of a default size and at a default position.

30. A method as in claim 27, comprising storing said gobo image in a matrix.

31. A method as in claim 28, comprising storing said gobo image in a matrix.

32. A method as in claim 31, further comprising changing the characteristics of the default gobo in the matrix.

33. A method as in claim 32, wherein said changing comprises scaling.

34. A method as in claim 32, wherein said changing comprises rotating.

35. A method as in claim 32, wherein said changing comprises applying an iris.

36. A method as in claim 32, wherein said changing comprises applying edge effects.

37. A method as in claim 32, wherein said changing comprises applying strobe information.

38. A method as in claim 32, wherein said changing comprises applying dimmer information to the matrix.

39. A method as in claim 32, wherein said changing comprises moving the image within the matrix.

40. A method as in claim 32, wherein said changing comprises replicating the gobo by forming another to default gobo, thereby providing multiple gobos handled by the same record.

41. A method as in claim 40, further comprising rotating all of the gobos.

42. A method as in claim 27, wherein said switching comprises fading out said old gobo while fading in said new gobo.

43. An apparatus comprising:

a first control element allowing selecting one of a plurality of gobos; and

a second control element allowing selection of a number of times to replicate a selected gobo type.

44. An apparatus as in claim 43, further comprising a memory storing image information about the gobo.

45. An apparatus as in claim 44, further comprising a memory storing a matrix representation of the gobo.

46. An apparatus as in claim 44, wherein said image memory stores a default image of a default gobo.

47. An apparatus as in claim 46, further comprising another control enabling modifying the default image.

48. A method comprising:

selecting a shape of a light beam to be projected;

forming a record for said shape; and

replicating said shape to provide more than one of said shapes, each of said more than one shape being handled by a single record.

49. A method as in claim 48, further comprising using said record to shape a beam of light.

50. A method as in claim 49, wherein the record includes a matrix which includes information about bits to be mapped to a digital gobo device.

51. A method as in claim 50, further comprising using a digital signal processor to carry out operations on the matrix, said operations including at least an operation of simulating a rotation of the image defined by the matrix.

52. A system as in claim 49, wherein the record further includes color information.

53. A method as in claim 48, wherein the record further includes information about a shape and size of an outside part of the beam being displayed.

54. A method as in claim 53, wherein said shape is an iris.

55. A method as in claim 48, further comprising commanding multiple gobos to spin in opposite directions.

56. A method as in claim 48, wherein said record defines two separate gobos.

57. A method of digitally cross-fading gobos in a stage light, comprising:

obtaining a first image to be used as a gobo;

obtaining a second image to be used as a second gobo; and

fading said first gobo out while fading said second gobo in.

58. A method as in claim 57, wherein said first gobo is stored at a first location, said second gobo is stored at a second location and further comprising pointing to said first location while said first gobo is being used, and pointing to said second location while said second gobo is being used.

59. A method as in claim 57, wherein said first and second gobos have different outer shapes.

60. A method comprising:

defining gobo information indicating a default gobo;

selecting said default gobo to create a default image, of a specified size, at a default location, said image indicative of the gobo; and

subsequently, after creating the default image, allowing the default image to be edited to form an edited gobo.

61. A method as in claim 60, further comprising projecting a light beam in the shape of the edited gobo.

62. A method as in claim 61, wherein said editing comprises changing the position of the gobo.

63. A method as in claim 62, wherein said editing comprises changing a size of the gobo.

64. A method as in claim 61, wherein said editing comprises changing a size of the gobo.

65. A method as in claim 61, further comprising replicating the default gobo to form an additional gobo.

66. A method as in claim 61, wherein said gobos are formed in memory within a matrix structure, and said editing comprises editing the matrix.

67. A method as in claim 66, wherein said editing comprises shifting the gobo to a specified position in the matrix, and forming another gobo at the default position.

68. A method as in claim 67, wherein said editing is carried out with a digital signal processor.

69. A method as in claim 61, wherein said gobos have multiple definable parameters, and said editing comprises allowing change of any of said parameters.

70. A method comprising:

obtaining a first image to be used as a gobo for projection of a light beam;

replicating said first image, to form multiple images based on the same base image; and

using the multiple images to change a shape of the light beam.

71. A method as in claim 70, further comprising allowing any of the multiple images to be edited.

72. A method as in claim 71, wherein the editing comprises scaling.

73. A method as in claim 71, wherein the editing comprises shifting in position.

74. A method as in claim 71, wherein the image is represented by information in a matrix in memory, and wherein said editing comprises editing the matrix.

75. A method as in claim 74, further comprising rotating the image forming the gobo using matrix arithmetic to rotate the matrix.

76. A method as in claim 74, further comprising shifting a position of the image in the matrix using matrix arithmetic.

77. A record format, comprising:

a first part which indicates a type of gobo, wherein a first type of gobo has a different overall shape than a second type of gobo; and

a second part which indicates controls to said first gobo, including at least a capability to control a moving part of said gobo, to scale a size of said gobo and to rotate said gobo.

78. A format as in claim 77, wherein said type of gobo includes an indication of whether the gobo is fixed, or has one or more controls.

79. A format as in claim 78, wherein said one or more controls comprises an indication that the gobo has only a single control or has a double control.

80. A method, comprising:

forming a first record which indicates a first image to be used as a first gobo that shapes output light;

forming a second record which indicates a second image to be used as a second gobo that shapes output light; and

using said first and second images to form a composite shape of the output light.

81. A method as in claim 80, further comprising allowing said first and second image to be controlled totally separately.

82. A method as in claim 81, wherein said control comprises controlling said first gobo image to rotate in a first direction and controlling said second gobo image to rotate in a second direction opposite from said first direction.

83. A method as in claim 81, further comprising controlling a brightness of said first and second images at their overlap.

84. A method as in claim 83, wherein said controlling comprises calculating a relative brightness of the gobos at said overlap.

85. A method as in claim 83, wherein one of said gobos is visible at an intersection between said first and second gobos.

86. A method as in claim 83, wherein one of said gobos is brighter than the other of said gobos at the intersection to enable said one gobo to be seen over the other gobo.

87. A method comprising:

defining an image file representing a shape of light to be displayed;

changing said image file in a way to simulate depth and perspective to the image; and

displaying light in a shape represented by the changed shape.

88. A method as in claim 87, wherein said changing comprises rotating the shape around an axis thereof.

89. A method as in claim 88, wherein said rotating in a way such that one part of the gobo appears to be move toward the viewer and another part of the gobo appears to move away from the viewer.

90. A method as in claim 88, wherein said rotating comprises rotating around the Z axis.

91. An apparatus, comprising:

a first pixel controllable device, which accepts commands indicating, on a pixel level, whether to pass light or not;

a digital signal processor, operating to calculate an image information for said pixel controllable device; and

a memory, storing information about at least a plurality of gobo types, wherein said digital signal processor selects one of said gobo types from memory to display a gobo.

92. An apparatus as in claim 91, further comprising a communication device, receiving information indicative of a gobo to be selected.

93. An apparatus as in claim 92, wherein said digital signal processor is operative to retrieve said gobo information from said memory responsive to said information.

94. An apparatus as in claim 92, further comprising an operation console, coupled to said communication device, said operation console including controls for editing an image representing said gobo.

95. An apparatus as in claim 94, wherein said controls include encoders.

96. An apparatus as in claim 94, wherein said gobo type is represented by a number.

97. An apparatus as in claim 96, wherein said gobo type also include gobo edit information which indicates parameters of the gobo.

98. An apparatus as in claim 97, wherein said gobo edit information includes at least information indicative of a size of the gobo.

99. An apparatus as in claim 97, wherein said gobo edit information includes at least information indicative of a position of the gobo.

100. A device as in claim 97, wherein said gobo edit information includes at least information indicative of a rotation of the gobo.

101. A method comprising:

defining a first image which defines a first shape of light to be projected;

defining a second image which defines a second shape of light to be projected; and

defining a timing of moving between said first and second shapes.

102. A method as in claim **101**, wherein said first and second images comprise a first basic shape, and first and second parameters for said first basic shape.

103. A method as in claim **102**, wherein said first basic shape includes a ring and said first and second parameters include a thickness of the ring.

104. A method as in claim **101**, wherein said first and second images comprise a basic gobo shape with a moving part in a first position and a moving part in a second position.

105. A method as in claim **101**, further comprising displaying light that is shaped by said images.

106. A method as in claim **101**, wherein said first shape is a first polygon, and said second shape is a second polygon.

107. An apparatus, comprising:

a light controlling console which produces first values indicative of a type of light shape altering device, said first values being at a first location in a message sent by said light controlling console and which produces at least one second value indicative of said light shape; and

a remote lamp, located remote from said console, and receiving said first and second values, and which processes said first values, and uses the result of processing said first values to determine the meaning of said second value.

108. An apparatus as in claim **107**, wherein said second value represents size controls for said light shape.

109. An apparatus as in claim **107**, wherein said second value represents position controls for said light shape.

110. An apparatus as in claim **107**, wherein the message includes a plurality of parts including an indication of information about the light shape and control values packet along with the information about the light shape.

111. A system as in claim **107**, wherein said second values includes timing information.

* * * * *

EXHIBIT J

US006288828B1

(12) **United States Patent**
Hewlett

(10) **Patent No.: US 6,288,828 B1**
(45) **Date of Patent: Sep. 11, 2001**

(54) **PROGRAMMABLE LIGHT BEAM SHAPE ALTERING DEVICE USING PROGRAMMABLE MICROMIRRORS**

(75) Inventor: **William E. Hewlett**, Sutton Coldfield (GB)

(73) Assignee: **Light and Sound Design Ltd.**, Birmingham (GB)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/359,064**

(22) Filed: **Jul. 21, 1999**

Related U.S. Application Data

(62) Division of application No. 08/926,237, filed on Sep. 10, 1997, now Pat. No. 5,984,465.

(51) **Int. Cl.**[7] **G02B 26/00**; G03B 21/00
(52) **U.S. Cl.** **359/291**; 359/292; 353/121
(58) **Field of Search** 359/290, 291, 359/292, 293, 295, 298, 223, 850, 896, 900; 353/28, 121

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,947,302		8/1990	Callahan	362/233
4,949,020		8/1990	Warren et al.	315/297
5,023,709		6/1991	Kita et al.	358/125
5,231,388		7/1993	Stoltz	340/701
5,325,473	*	6/1994	Monroe et al.	395/129
5,331,359	*	7/1994	Leclercq	353/28
5,386,250		1/1995	Guerinot	348/770
5,406,176		4/1995	Sugden	315/292
5,452,024		9/1995	Sampsell	348/291
5,467,146		11/1995	Huang et al.	348/743
5,583,688		12/1996	Hornbeck	359/291
5,647,662		7/1997	Ziegler et al.	352/294
5,940,204	*	8/1999	Hewlett	359/298

FOREIGN PATENT DOCUMENTS

0 399 496	11/1990	(EP) .
0 385 706	5/1995	(EP) .
0 662 773	6/1999	(EP) .
0 511 829	8/1999	(EP) .
2 267 788	12/1993	(GB) .
93/18620	9/1993	(WO) .

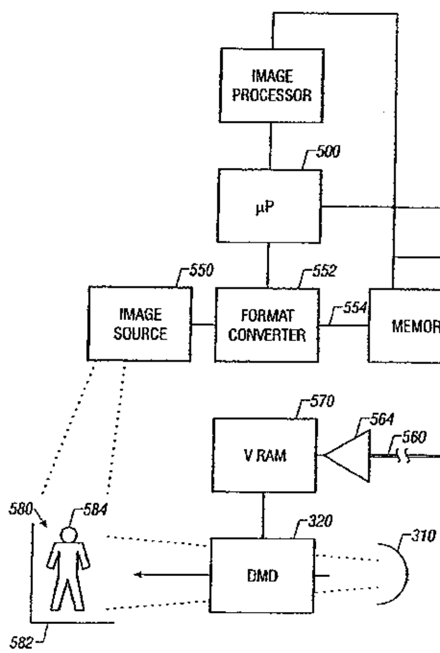
* cited by examiner

Primary Examiner—Ricky Mack
(74) *Attorney, Agent, or Firm*—Fish & Richardson P.C.

(57) **ABSTRACT**

A digital micromirror device ("DMD") is used to alter the shape of light that is projected onto a stage. The DMD selectively reflects some light, thereby shaping the light that is projected onto the stage. The control for the alteration is controlled by an image. That image can be processed, thereby carrying out image processing effects on the shape of the light that is displayed. One preferred application follows the shape of the performer and illuminates the performer using a shape that adaptively follows the performer's image. This results in a shadowless follow spot.

22 Claims, 10 Drawing Sheets



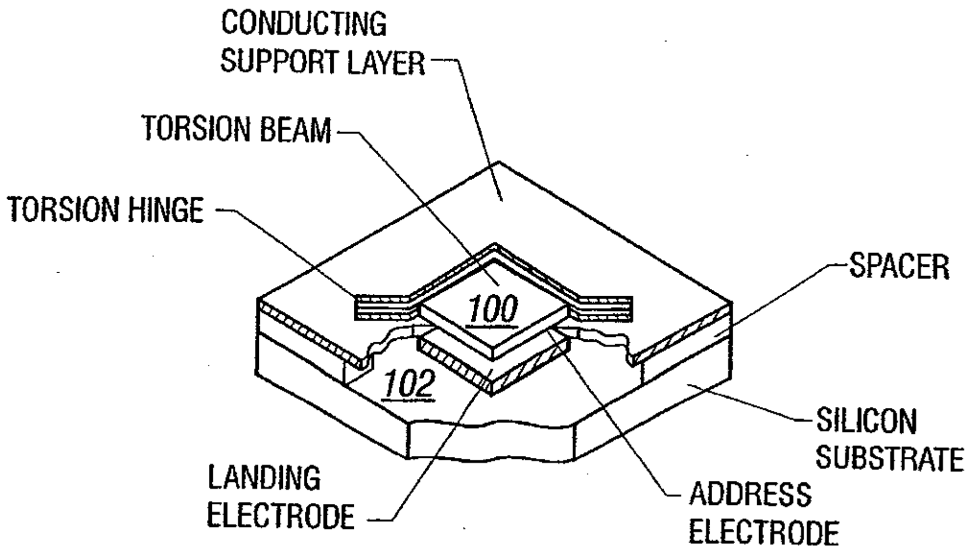

CONDUCTING
SUPPORT LAYER
TORSION BEAM
TORSION HINGE
100
102
SPACER
SILICON
SUBSTRATE
LANDING
ELECTRODE
ADDRESS
ELECTRODE

FIG 1

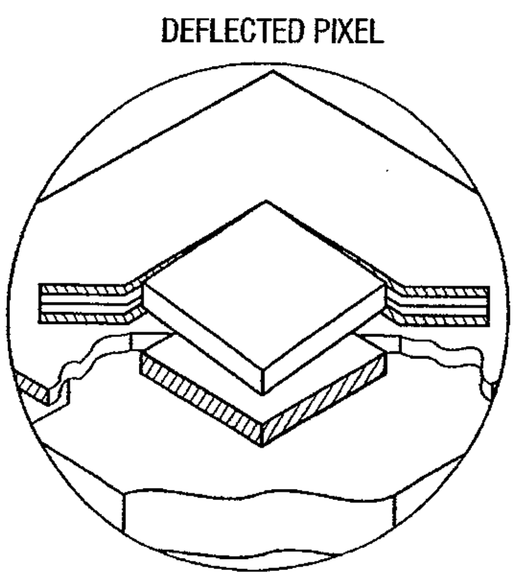
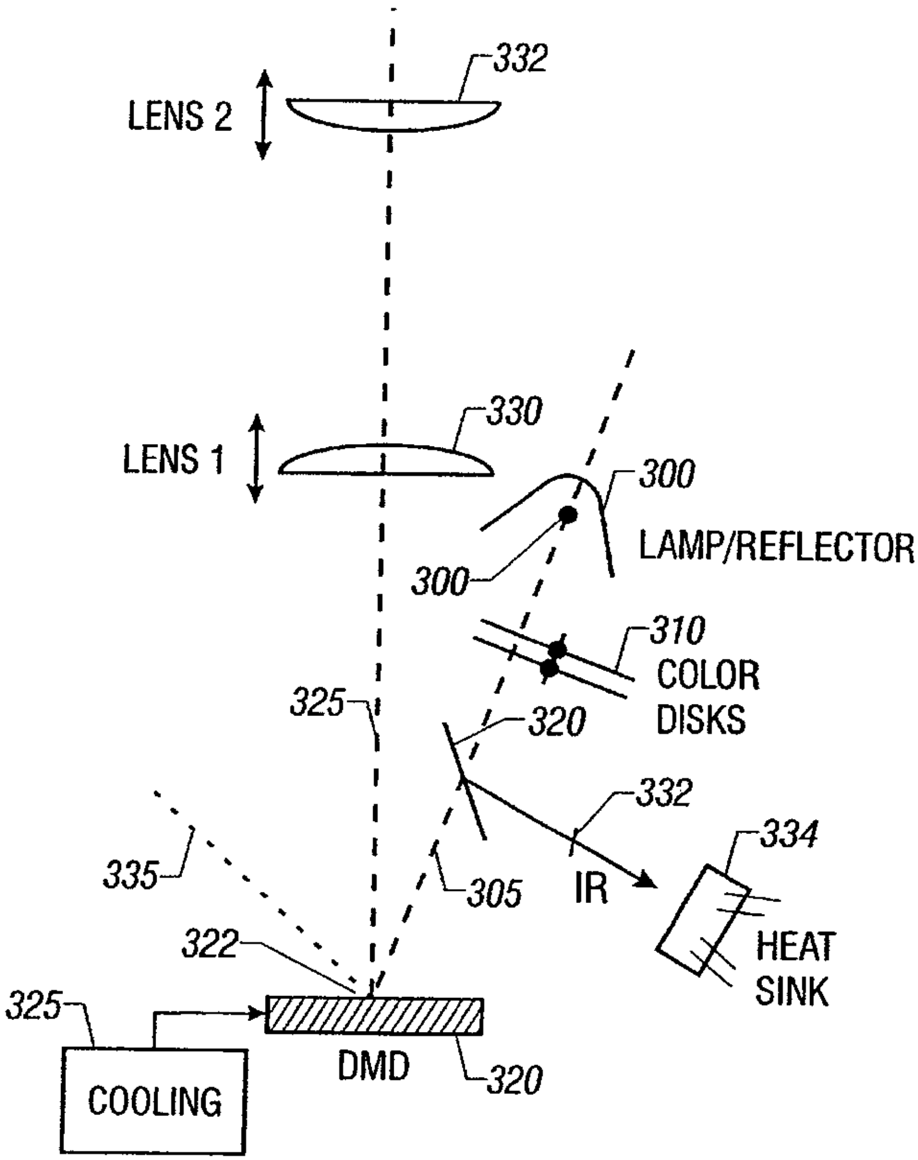

DEFLECTED PIXEL

FIG 2



332
LENS 2
330
LENS 1
300
LAMP/REFLECTOR
300
310
COLOR
DISKS
325
320
332
334
335
305
IR
HEAT
SINK
322
325
DMD
320
COOLING

FIG. 3

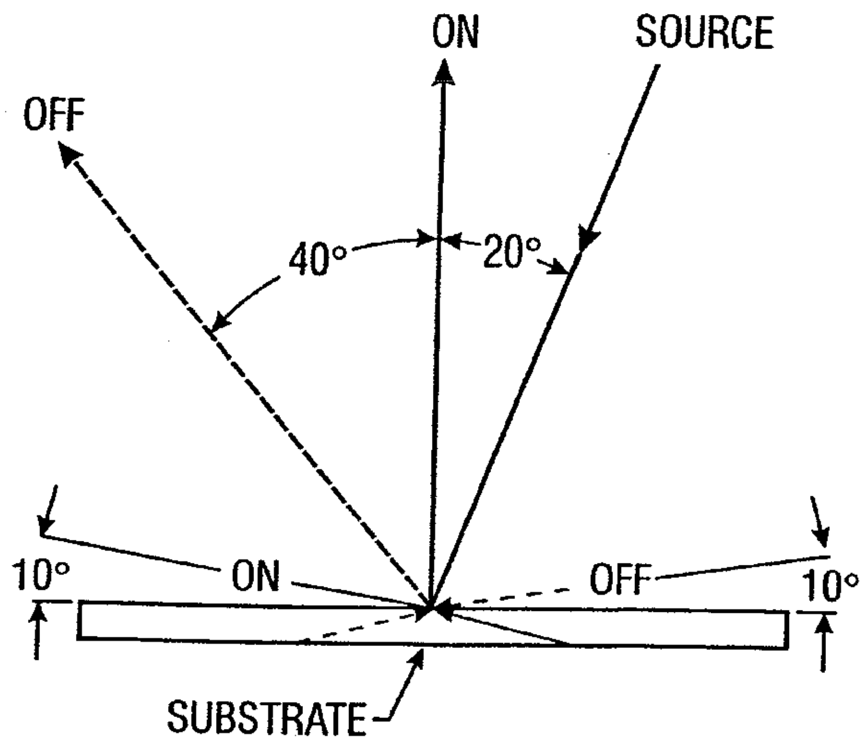


FIG. 4

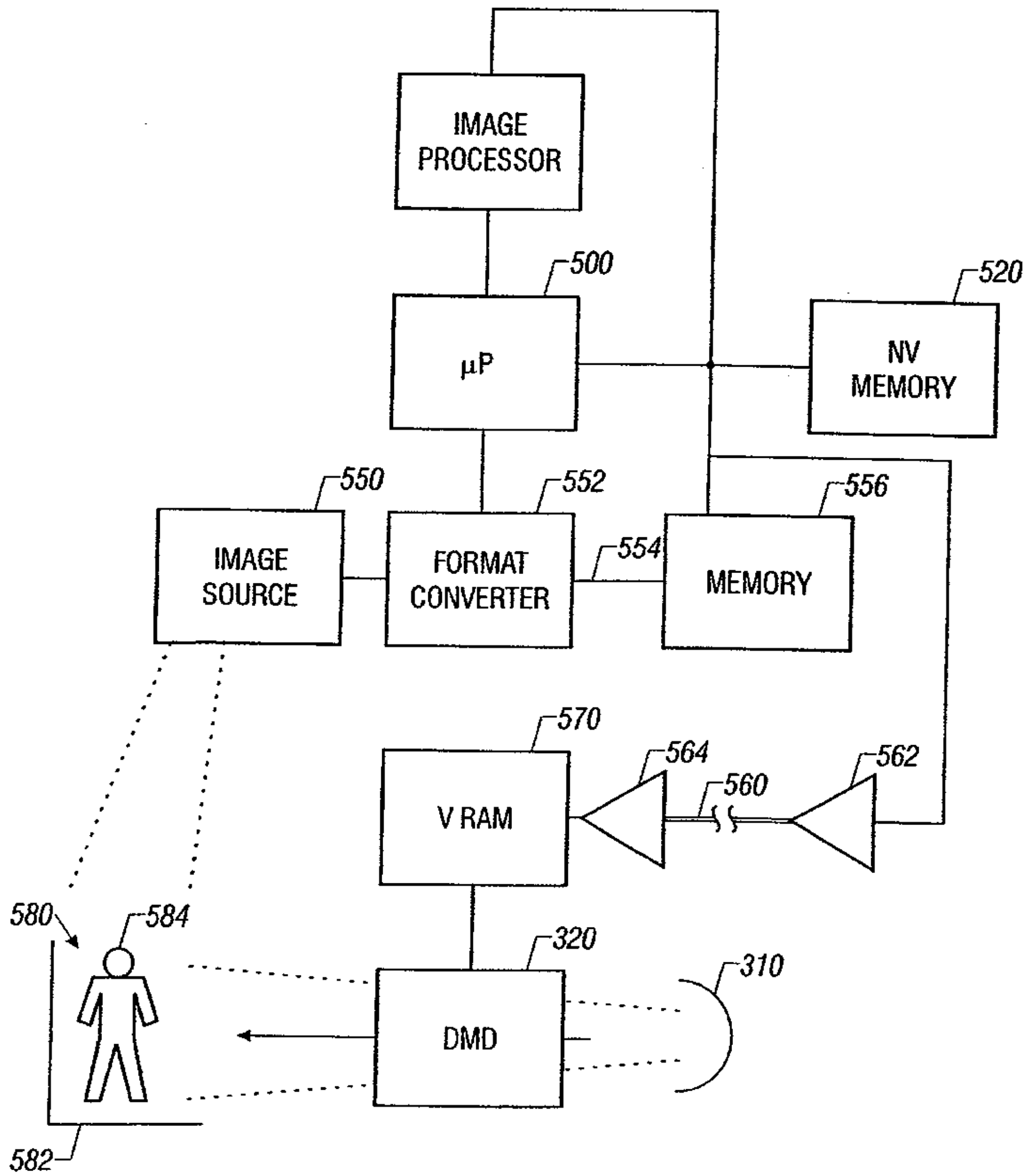
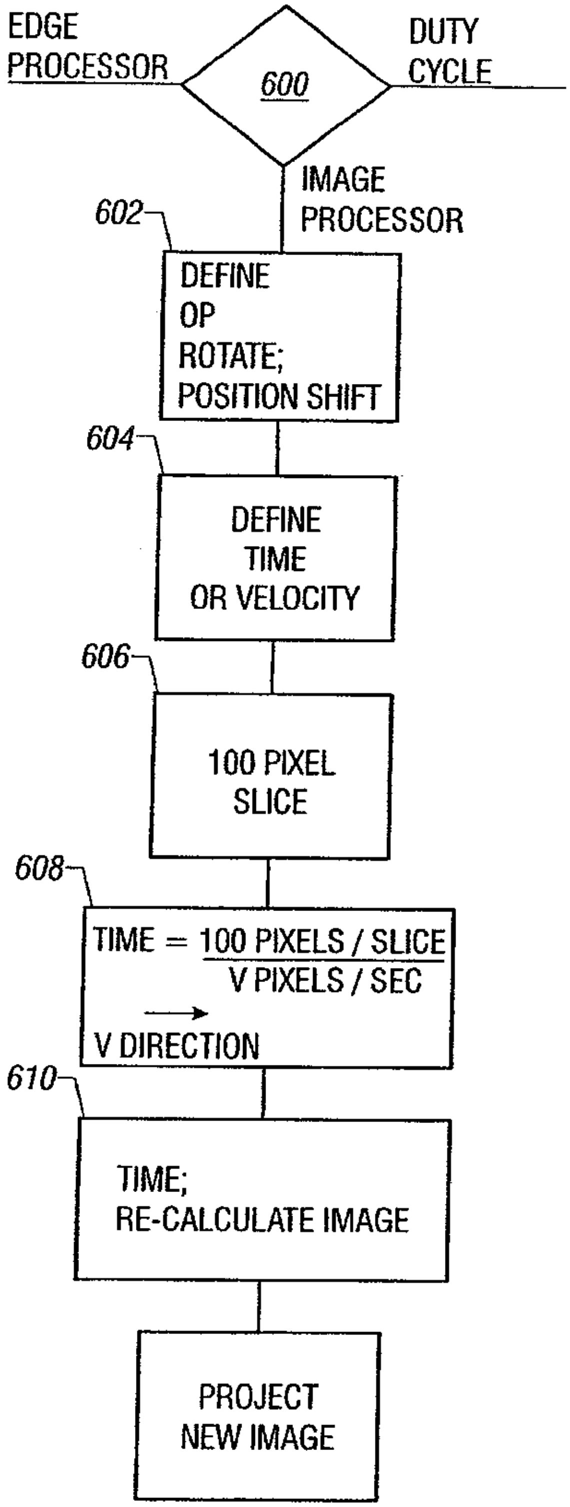


IMAGE PROCESSOR
500
μP
520
NV MEMORY
550
IMAGE SOURCE
552
FORMAT CONVERTER
554
556
MEMORY
570
V RAM
564
560
562
580
584
320
310
DMD
582

FIG. 5



EDGE PROCESSOR
600
DUTY CYCLE
IMAGE PROCESSOR
602
DEFINE
OP
ROTATE;
POSITION SHIFT
604
DEFINE
TIME
OR VELOCITY
606
100 PIXEL
SLICE
608
TIME = 100 PIXELS / SLICE / V PIXELS / SEC
V DIRECTION
610
TIME;
RE-CALCULATE IMAGE
PROJECT
NEW IMAGE

FIG. 6

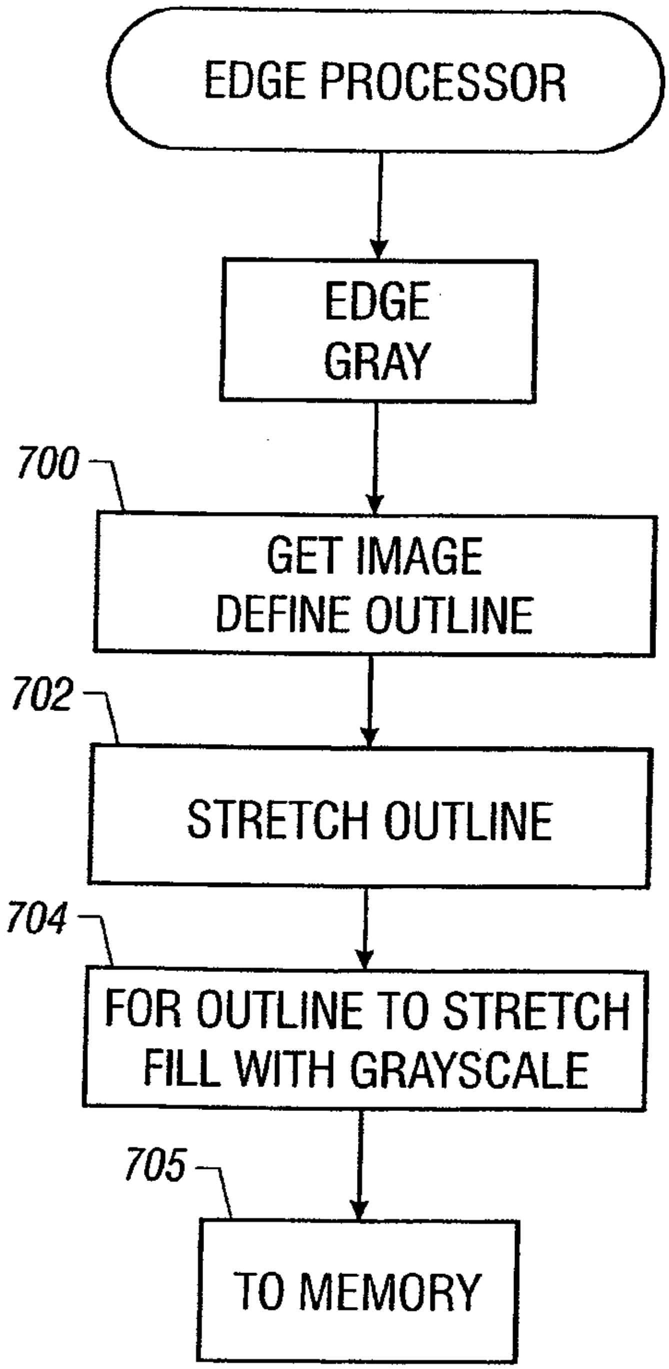

EDGE PROCESSOR
EDGE GRAY
700
GET IMAGE DEFINE OUTLINE
702
STRETCH OUTLINE
704
FOR OUTLINE TO STRETCH FILL WITH GRAYSCALE
705
TO MEMORY

FIG. 7

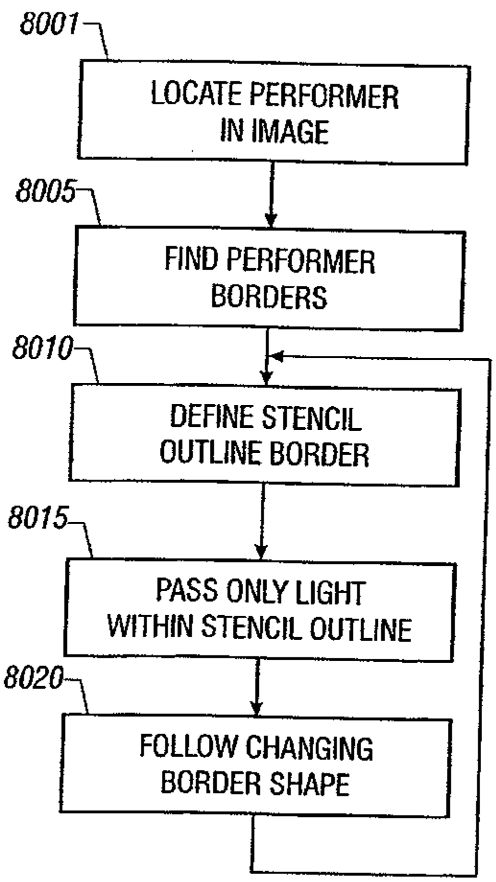

8001
LOCATE PERFORMER IN IMAGE
8005
FIND PERFORMER BORDERS
8010
DEFINE STENCIL OUTLINE BORDER
8015
PASS ONLY LIGHT WITHIN STENCIL OUTLINE
8020
FOLLOW CHANGING BORDER SHAPE

FIG. 8A

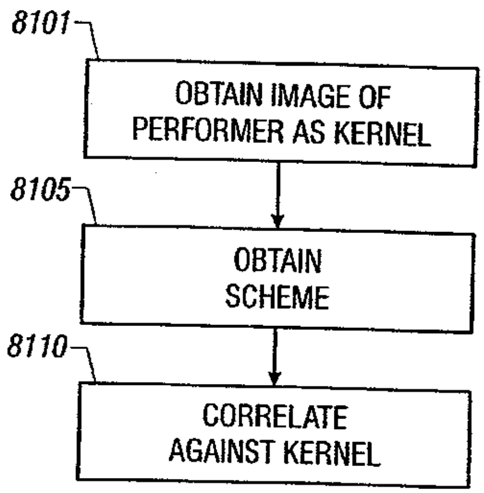

8101
OBTAIN IMAGE OF PERFORMER AS KERNEL
8105
OBTAIN SCHEME
8110
CORRELATE AGAINST KERNEL

FIG. 8B

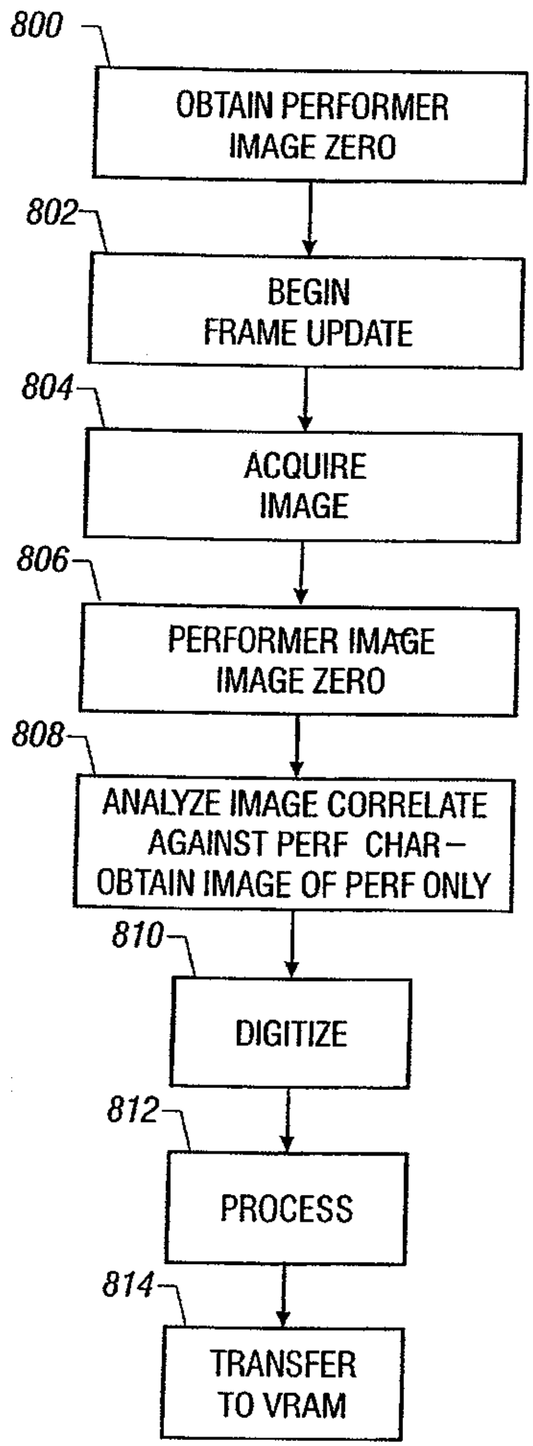

800
OBTAIN PERFORMER IMAGE ZERO
802
BEGIN FRAME UPDATE
804
ACQUIRE IMAGE
806
PERFORMER IMAGE IMAGE ZERO
808
ANALYZE IMAGE CORRELATE AGAINST PERF CHAR – OBTAIN IMAGE OF PERF ONLY
810
DIGITIZE
812
PROCESS
814
TRANSFER TO VRAM

FIG. 8C

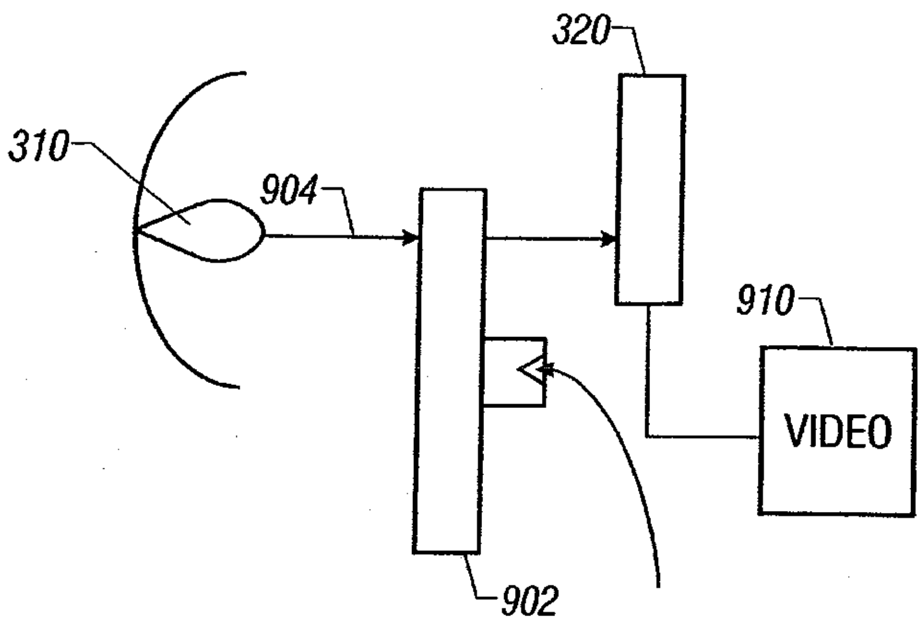


FIG. 9A

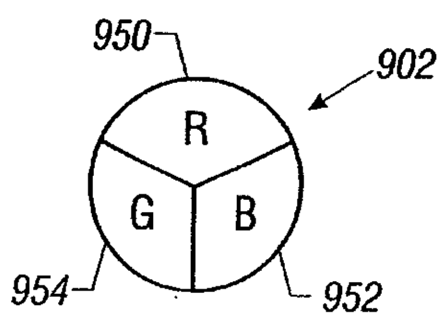


FIG. 9B

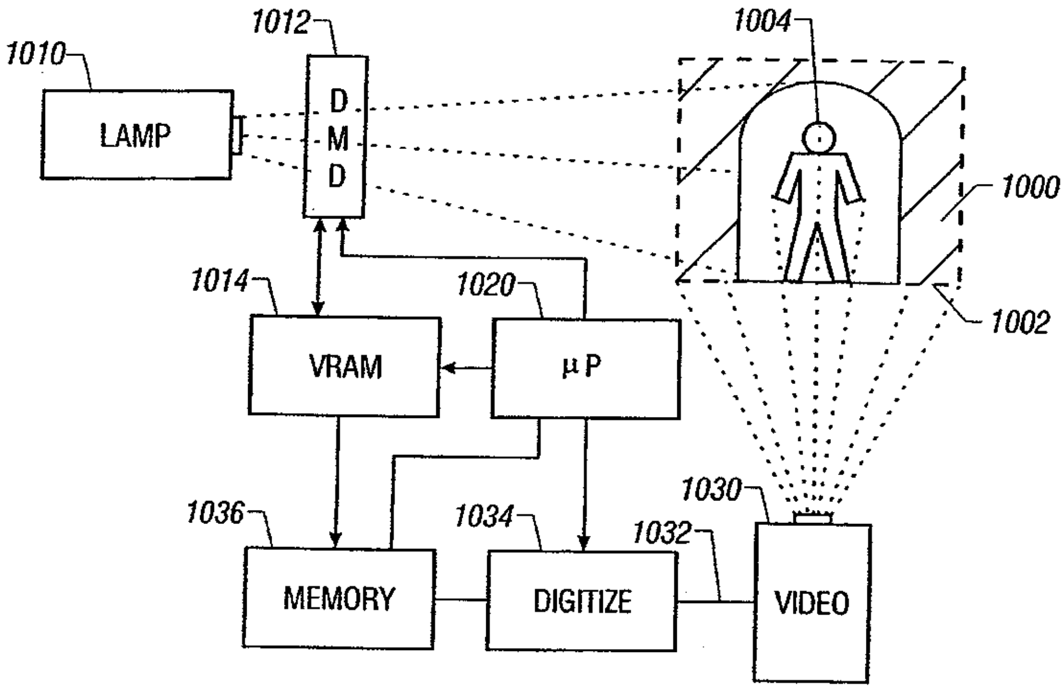

1010
1012
1004
LAMP
D
M
D
1000
1002
1014
1020
VRAM
µP
1030
1036
1034
1032
MEMORY
DIGITIZE
VIDEO

FIG. 10

PROGRAMMABLE LIGHT BEAM SHAPE ALTERING DEVICE USING PROGRAMMABLE MICROMIRRORS

This is a divisional of U.S. application Ser. No. 08/926,237, filed Sep. 10, 1997, now U.S. Pat. No. 5,984,465.

FIELD OF THE INVENTION

The present invention relates to a programmable light beam shaping device. More specifically, the present invention teaches a control system and micromirror device which can alter the shape of light beams passing therethrough, and provide various effects to those shaped light beams.

BACKGROUND OF THE INVENTION

It is known in the art to shape a light beam. This has typically been done using an element known as a gobo. A gobo element is usually embodied as either a shutter or an etched mask. The gobo shapes the light beam like a stencil in the projected light.

Gobos are simple on/off devices: they allow part of the light beam to pass, and block other parts to prevent those other parts from passing. Hence mechanical gobos are very simple devices. Modern laser-etched gobos go a step further by providing a gray scale effect.

Typically multiple different gobo shapes are obtained by placing the gobos are placed into a cassette or the like which is rotated to select between the different gobos. The gobos themselves can also be rotated within the cassette, using the techniques, for example, described in U.S. Pat. Nos. 5,113,332 and 4,891,738.

All of these techniques, have the drawback that only a limited number of gobo shapes can be provided. These gobo shapes must be defined in advance. There is no capability to provide any kind of gray scale in the system. The resolution of the system is also limited by the resolution of the machining. This system allows no way to switch gradually between different gobo shapes. In addition, moving between one gobo and another is limited by the maximum possible mechanical motion speed of the gobo-moving element.

Various patents and literature have suggested using a liquid crystal as a gobo. For example, U.S. Pat. No. 5,282,121 describes such a liquid crystal device. Our own pending patent application also so suggests. However, no practical liquid crystal element of this type has ever been developed. The extremely high temperatures caused by blocking some of this high intensity beam produce enormous amounts of heat. The projection gate sometimes must block beams with intensities in excess of 10,000 lumens and sometimes as high as 2000 watts. The above-discussed patent applications discuss various techniques of heat handling. However, because the light energy is passed through a liquid crystal array, some of the energy must inevitably be stored by the liquid crystal. Liquid crystal is not inherently capable of storing such heat, and the phases of the liquid crystal, in practice, may be destabilized by such heat. The amount of cooling required, therefore, has made this an impractical task. Research continues on how to accomplish this task more practically.

It is an object of the present invention to obviate this problem by providing a digital light beam shape altering device, e.g. a gobo, which operates completely differently than any previous device. Specifically, this device embodies the inventor's understanding that many of the heat problems in such a system are obviated if the light beam shape altering

device would selectively deflect, instead of blocking, the undesired light.

The preferred mode of the present invention uses a digitally-controlled micromirror semiconductor device. However, any selectively-controllable multiple-reflecting element could be used for this purpose. These special optics are used to create the desired image using an array of small-sized mirrors which are movably positioned. The micromirrors are arranged in an array that will define the eventual image. The resolution of the image is limited by the size of the micromirrors: here 17 μm on a side.

The mirrors are movable between a first position in which the light is directed onto the field of a projection lens system, or a second position in which the light is deflected away from the projection lens system. The light deflected away from the lens will appear as a dark point in the resulting image on the illuminated object. The heat problem is minimized according to the present invention since the micromirrors reflect the unwanted light rather than absorbing it. The absorbed heat is caused by the quantum imperfections of the mirror and any gaps between the mirrors.

A digital micromirror integrated circuit is currently manufactured by Texas Instruments Inc., Dallas, Tex., and is described in "an overview of Texas Instrument digital micromirror device (DMD) and its application to projection displays". This application note describes using a digital micromirror device in a television system. Red, green and blue as well as intensity grey scales are obtained in this system by modulating the micromirror device at very high rates of speed. The inventor recognized that this would operate perfectly to accomplish his objectives.

It is hence an object of the present invention to adapt such a device which has small-sized movable, digitally controllable mirrors which have positions that can be changed relative to one another, to use as a light beam shape altering device in this stage lighting system.

It is another object of the present invention to use such a system for previously unheard-of applications. These applications include active simulation of hard or soft beam edges on the gobo. It is yet another application of the present invention to allow gobo cross-fading using time control, special effects and morphing.

It is yet another object of the present invention to form a stroboscopic effect with variable speed and intensity in a stage lighting system. This includes simulation of a flower strobe.

Yet another object of the present invention is to provide a multiple colored gobo system which can have split colors and rotating colors.

It is yet another object of the present invention to carry out gobo rotation in software, and to allow absolute position and velocity control of the gobo rotation using a time slicing technique.

Another objective is to allow concentric-shaped images and unsupported images.

It is yet another object of the invention to provide a control system for the micromirror devices which allows such operation.

Yet another particularly preferred system is a shadowless follow spot, which forms an illuminating beam which is roughly of the same shape as the performer, and more preferably precisely the same as the performer. The beam shape of the beam spot also tracks the performer's current outline. The spot light follows the performer as it lights the performer. This action could be performed manually by an

operator or via an automated tracking system, such as Wybron's autopilot.

Since the beam does not overlap the performer's body outline, it does not cast a shadow of the performer.

BRIEF DESCRIPTION OF THE DRAWINGS

These and other objects will be readily understood with reference to the accompanying drawings, in which:

FIG. **1** shows a single pixel mirror element of the preferred mode, in its first position;

FIG. **2** shows the mirror element in its second position;

FIG. **3** shows the mirror assembly of the present invention and its associated optics;

FIG. **4** shows more detail about the reflection carried out by the DMD of the present invention;

FIG. **5** shows a block diagram of the control electronics of the present invention;

FIG. **6** shows a flowchart of a typical operation of the present invention;

FIG. **7** shows a flowchart of operation of edge effects operations;

FIG. **8**A shows a flowchart of a first technique of following a performer on stage;

FIG. **8**B shows a flowchart of a correlation scheme;

FIG. **8**C shows a flowchart of another correlation scheme;

FIG. **9**A shows a block diagram of a color projection system of the present invention;

FIG. **9**B shows a color wheel of the present invention; and

FIG. **10** shows a block diagram of the shadowless follow spot embodiment.

DESCRIPTION OF THE PREFERRED EMBODIMENT

The preferred embodiment herein begins with a brief description of controllable mirror devices, and the way in which the currently-manufactured devices operate.

Work on semiconductor-based devices which tune the characteristics of light passing therethrough has been ongoing since the 1970's. There are two kinds of known digital micromirror devices. A first type was originally called the formal membrane display. This first type used a silicon membrane that was covered with a metalized polymer membrane. The metalized polymer membrane operated as a mirror.

A capacitor or other element was located below the metalized element. When the capacitor was energized, it attracted the polymer membrane and changed the direction of the resulting reflection.

More modern elements, however, use an electrostatically deflected mirror which changes in position in a different way. The mirror of the present invention, developed and available from Texas Instruments, Inc. uses an aluminum mirror which is sputter-deposited directly onto a wafer.

The individual mirrors are shown in FIG. **1**. Each individual mirror includes a square mirror plate **100** formed of reflective aluminum cantilevered on hollow aluminum post **102** on flexible aluminum beams. Each of these mirrors **100** have two stop positions: a landing electrode, which allows them to arrive into a first position shown in FIG. **2**, and another electrode against which the mirror rests when in its non-deflected position. These mirrors are digital devices in the sense that there two "allowable" positions are either in a first position which reflects light to the lens and hence to the illuminated object, and a second position where the light is reflected to a scattered position. Light scattering (i.e. selective light reflection) of this type could also be done with other means, i.e. selectively polarizable polymers, electronically-controlled holograms, light valves, or any other means.

The operation of the dark field projection optics which is used according to the preferred micromirror device is shown in FIG. **3**. The two bi-stable positions of the preferred devices are preferably plus or minus 10% from the horizontal.

An incoming illumination bundle **303** is incident at an arc of less than 20° on the digital micromirror device **220**. The illumination bounces off the mirrors in one of two directions **230** or **232** depending on the mirror position. In the first direction **302**, the position we call "on", the information is transmitted in the 0° direction **300** towards lens **302** which focuses the information to the desired location **304**. In the second direction of the mirror, the position we call "off", the information is deflected away from the desired location to the direction **306**.

The human eye cannot perceive actions faster than about 1/30 second. Importantly, the mirror transit time from tilted left to tilted right is on the order of 10 μs. This allows the pixels to be changed in operation many orders of magnitude faster than the human eye's persistence of vision.

Light source **310** used according to the present invention is preferably a high intensity light source such as a xenon or metal halide bulb of between 600 and 1000 watts. The bulb is preferably surrounded by a reflector of the parabolic or ellipsoidal type which directs the output from bulb **300** along a first optical incidence path **305**.

The preferred embodiment of the invention provides a color cross-fading system **315**, such as described in my U.S. Pat. No. 5,426,476. Alternately, however, any other color changing system could be used. This cross-fading system adjusts the color of the light. The light intensity may also be controlled using any kind of associated dimmer; either electronic, mechanical or electromechanical means. More preferably, the DMD **320** could be used to control beam intensity as described herein.

The light beam projected **310** along path **305** is incident to the digital light altering device embodied as DMD **320**, at point **322**. The DMD allows operations between two different states. When the mirror in the DMD is pointed to the right, the right beam is reflected along path **325** to projection/zoom lens combination **330, 332**. The zoom lens combination **330, 332** is used to project the image from the DMD **320** onto the object of illumination, preferably a stage. The size and sharpness quality of the image can therefore be adjusted by repositioning of the lens. When the mirror is tilted to the right, the light beam is projected along the light path **335**, away from projection lens **330/332**. The pixels which have light beams projected away from the lens appear as dark points in the resulting image. The dark spots are not displayed on the stage.

This DMD system reflects information from all pixels. Hence, minimal energy is absorbed in the DMD itself or any of the other optics. The device still may get hot, however not nearly as hot as the liquid crystal gobos. Cooling **325** may still be necessary. The DMDs can be cooled using any of the techniques described in (Bornhorst LCD), or by a heat sink and convection, or by blowing cold air from a refrigeration unit across the device. More preferably, a hot or cool mirror can be used in the path of the light beam to reflect infrared out of the light beam to minimize the transmitted heat. FIG.

3 shows hot mirror **330** reflecting infra red **332** to heat sink **334**. A cold mirror would be used with a folded optical path.

This basic system allows selecting a particular aperture shape with which to which pass the light. That shape is then defined in terms of pixels, and these pixels are mapped to DMD **320**. The DMD selectively reflects light of the properly-shaped aperture onto the stage. The rest of the light is reflected away.

The micromirror can be switched between its positions in approximately 10 μs. A normal time for frame refresh rate, which takes into account human persistence of vision, is 1/60th of a second or 60 hertz. Various effects can be carried out by modulating the intensity of each mirror pixel within that time frame.

The monolithic integration which is being formed by Texas Instruments includes associated row and column decoders thereon. Accordingly, the system of the present invention need not include those as part of its control system.

Detailed operation of DMD **320** is shown in FIG. **4**. The source beam is input to the position **322** which transmits the information either towards the stage along path **325** or away from the stage along path **335**.

The various effects which are usable according to the present invention include automatic intensity dimming, use of a "shadowless follow spot", hard or soft beam edges, shutter cut simulation, gobo cross fading, gobo special effects, stroboscopic effects, color gobos, rotating gobos including absolute position and velocity control, and other such effects and combinations thereof. All of these effects can be controlled by software running on the processor device. Importantly, the characteristics of the projected beam (gobo shape, color etc) can be controlled by software. This enables any software effect which could be done to any image of any image format to be done to the light beam. The software that is used is preferably image processing software such as Adobe photoshop™ Kai's power tools™ or the like which are used to manipulate images. Any kind of image manipulation can be mapped to the screen. Each incremental changes to the image can be mapped to the screen as it occurs.

Another important feature of the gobo is its ability to project unconnected shapes that cannot be formed by a stencil. An example is two concentric circles. A concentric circle gobo needs physical connection between the circles. Other unconnected shapes which are capable of rendering as an image can also be displayed.

The effects carried out by the software are grouped into three different categories: an edge effects processing; an image shape processing; and a duty cycle processing.

The overall control system is shown in block diagram form in FIG. **5**. Microprocessor **500** operates based on a program which executes, inter alia, the flowchart of FIG. **6**. The light shape altering operates according to a stencil outline. This stencil outline can be any image or image portion. An image from image source **552** is input to a format converter **552** which converts the image from its native form into digital image that is compatable with storage on a computer. The preferred digital image formats include a bitmap format or compressed bitmap form such as the GIF, JPEG, PCX format (1 bit per pixel) file, a "BMP" file (8 bits/pixel B/W or 24 bits/pixel color) or a geometric description (vectorized image). Moving images could also be sent in any animation format such as MPEG or the like. It should be understood that any image representation format could be used to represent the image, and that any of these

representations can be used to create information that can modify reflecting positions of the array of reflecting devices. The present specification uses the term "digital representation" to generically refer to any of these formats that can be used to represent an image, and are manipulable by computers.

Image **554** is input into a working memory **556**. BMP format represents each "pixel" picture element of the image by a number of bits. A typical gray scale bit map image has 8 bits representing each pixel. A colored image of this type has 8 bits representing each of red, green, and blue representations. This color representation is called a 24 -bit representation, since 24-bits are necessary for each pixel. The description herein will be given with reference to gray scale images although it should be understood that this system can also be used with color images by forming more detailed maps of the information. Bit maps are easiest to process, but extremely wasteful of storage space.

Each memory area, representing each pixel, therefore, has 8 bits therein. The memory **556** is 576×768 area, corresponding to the number of mirror elements in the preferred use.

This image is defined as image No. x, and can be stored in non-volatile memory **520** (e.g., flash RAM or hard disk) for later recall therefrom. An important feature of the present invention is that the images are stored electronically, and hence these images can also be electronically processed in real time using image processing software. Since the preferred mode of the present invention manipulates the image information in bitmap form, this image processing can be carried out in a very quick succession.

The image to be projected is sent, by processor **500**, over channel **560**, to VRAM **570**. Line driver **562** and line receiver **564** buffer the signal at both ends. The channel can be a local bus inside the lamp unit, or can be a transmission line, such as a serial bus. The image information can be sent in any of the forms described above.

Standard and commonly available image processing software is available to carry out many functions described herein. These include for example, morphing, rotating, scaling, edge blurring, and other operations that are described herein. Commercial image processing can use "Kai's Power Tools", "CorelDraw!", or "Morph Studio" for example. These functions are shown with reference to the flowchart of FIG. **6**.

Step **600** represents the system determining the kind of operation which has been requested: between edge processing, image processing, and duty cycle processing. The image processing operations will be defined first. Briefly stated, the image processing operations include rotation of the image, image morphing from image **1** to image **2**, dynamic control of image shape and special effects. Each of these processing elements can select the speed of the processing to effectively time-slice the image. The morphing of the present invention preferably synchronizes keyframes of the morph with desired time slices.

Step **602** defines the operation. As described above, this operation can include rotation, position shift, and the like. Step **604** defines the time or velocity of operation. This time can be ending time for all or part of the movement, or velocity of the movement. Note that all of the effects carried out in step **602** require moving some part of the image from one position to another.

Step **606** determine the interval of slicing, depending on the velocity. It is desireable to slice an appropriate amount such that the user does not see jerky motion. Ideally, in fact,

we could slice movement of the image one pixel at a time, but this is probably unnecessary for most applications. One hundred pixel slicing is probably sufficient for all applications. The pixel slices are selected at step **606**.

Step **608** calculates using the time or velocity entered at step **604** to determine the necessary time for operation based on the amount of position shift for rotation over **100** pixel slices. This is done as follows. Position shift, rotate, and sprite animation are all simple movements. In both, the points of the image which define the gobo shape move over time. It is important, therefore, to decide how much movement there is and how much time that movement will take. A rate of change of points or velocity is then calculated. Of course velocity need not be calculated if it has already been entered at step **604**.

Having velocity of movement and pixels per second, the time between slices is calculated using **100** pixels per slice divided by the velocity in pixels per second. The direction of movement is defined by this operation.

Therefore, the image is recalculated at step **610** for each time interval. This new image becomes the new gobo stencil at the new location. That is to say, the outline of the image is preferably used as the gobo—light within the image is passed, and light outside the image is blocked. In the color embodiment described herein, more sophisticated operations can be carried out on the image. For example, this is not limited to stencil images, and could include for example concentric circles or letter text with font selection.

At any particular time, the image in the VRAM **570** is used as the gobo stencil. This is carried out as follows. Each element in the image is a gray scale of 8-bits. Each 1/60th of a second is time-sliced into 256 different periods. Quite conveniently, the 8-bit pixel image corresponds to $2^8=256$.

A pixel value of 1 indicates that light at the position of the pixel will be shown on the stage. A pixel value of zero indicates that light at the position of the pixel will not be shown on the stage. Any gray scale value means that only part of the intensity pixel will be shown (for only part of the time of the 1/60th of a second time slice). Hence, each element in the memory is applied to one pixel of the DMD, e.g. one or many micromirrors, to display that one pixel on the stage.

When edge processing is selected at step **600**, control passes to the flowchart of FIG. **7**. The edge graying can be selected as either a gradual edge graying or a more abrupt edge graying. This includes one area of total light, one area of only partial light, and one area of no light. The intensity of the gray scaled outline is continuously graded from full image transmission to no image transmission. The intensity variation is effected by adjusting the duty cycle of the on and off times.

Step **700** obtains the image and defines its outlines. This is carried out according to the present invention by determining the boundary point between light transmitting portions (1's) and light blocking portions (0's). The outline is stretched in all directions at step **702** to form a larger but concentric image—a stretched image.

The area between the original image and the stretched image is filled with desired gray scale information. Step **704** carries this out for all points which are between the outline and the stretch image.

This new image is sent to memory **570** at step **706**. As described above, the image in the memory is always used to project the image-shaped information. This uses standard display technology whereby the display system is continually updated using data stored in the memory.

The duty cycle processing in the flowchart of FIG. **6** is used to form strobe effects and/or to adjust intensity. In both cases, the image is stored in memory and removed from memory at periodic intervals. This operation prevents any light from being projected toward the stage at those intervals, and is hence referred to as masking. When the image is masked, all values in the memory become zero, and hence this projects all black toward the source. This is done for a time which is shorter than persistence of vision, so the information cannot be perceived by the human eye. Persistence of vision averages the total light impinging on the scene. The eye hence sees the duty cycle processing as a different intensity.

The stroboscopic effect turns on and off the intensity, ranging from about 1 Hz to 24 Hz. This produces a strobe effect.

These and other image processing operations can be carried out: (1) in each projection lamp based on a prestored or downloaded command; (2) in a main processing console; or (3) in both.

Another important aspect of the invention is based on the inventor's recognition of a problem that has existed in the art of stage lighting. Specifically, when a performer is on the stage, a spotlight illuminates the performer's area. However, the inventor of the present invention recognized a problem in doing this. Specifically, since we want to see the performer, we must illuminate the performer's area. However, when we illuminate outside the performer's area, it casts a shadow on the stage behind the performer. In many circumstances, this shadow is undesirable.

It is an object of this embodiment to illuminate an area of the stage confined to the performer, without illuminating any location outside of the performer's area. This is accomplished according to the present invention by advantageous processing structure which forms a "shadowless follow spot". This is done using the basic block diagram of FIG. **10**.

The preferred hardware is shown in FIG. **10**. Processor **1020** carries out the operations explained with reference to the following flowcharts which define different ways of following the performer. In all of these embodiments, the shape of the performer on the stage is determined. This can be done by (1) determining the performer's shape by some means, e.g. manual, and following that shape; (2) correlating over the image looking for a human body shape; (3) infra red detection of the performer's location followed by expanding that location to the shape of the performer; (4) image subtraction; (5) detection of special indices on the performer, e.g. an ultrasonic beacon, or, any other technique even manual following of the image by, for example, an operator following the performer's location on a screen using a mouse.

FIG. **8A** shows a flowchart of (1) above. At step **8001**, the performer is located within the image. The camera taking the image is preferably located at the lamp illuminating the scene in order to avoid parallax. The image can be manually investigated at each lamp or downloaded to some central processor for this purpose.

Once identified, the borders of the performer are found at **8005**. Those borders are identified, for example, by abrupt color changes near the identified point. At step **8010**, those changes are used to define a "stencil" outline that is slightly smaller than the performer at **8010**. That stencil outline is ued as a gobo for the light at **8015**.

The performer continues to move, and at **8020** the processor follows the chanting border shape. The changing border shape produces a new outline which is fed to **8010** at which time a new gobo stencil is defined.

Alternative (2) described above is a correlation technique. A flowchart of this operation is shown in FIG. **8B**. At step **8101**, the camera obtains an image of the performer, and the performer is identified within that image. That image issued as a kernel for further later correlation. The entire scene is obtained at step **8105**. The whole scene is correlated against the kernel at **8110**. This uses known image processing techniques.

The above can be improved by (3), wherein infra red detection gives the approximate area for the performer.

As explained in previous embodiments, the DMD is capable of updating its position very often: for example, 10^6 times a second. This is much faster than any real world image can move. Thirty times a second would certainly be sufficient to image the performer's movements. Accordingly, the present invention allows setting the number of frame updates per second. A frame update time of 30 per second is sufficient for most applications. This minimizes the load on the processor, and enables less expensive image processing equipment to be used.

FIG. **8C** shows the image subtracting technique.

First, we must obtain a zeroing image. Therefore, the first step at step **800**, is to obtain an image of the stage without the performer(s) thereon. This zero image represents what the stage will look like when the performers are not there.

Between processing iterations, the processor can carry out other housekeeping tasks or can simply remain idle.

Step **802** represents the beginning of a frame update. An image is acquired from the video camera **550** at step **804**. The image is still preferably arranged in units of pixels, with each pixel including a value of intensity and perhaps red, green, and blue for that pixel.

At step **806** subtracts the current image from the zeroed image. The performer image that remains is the image of the performer(s) and other new elements on the stage only. The computer determines at this time which part of that image we want to use to obtain the shadowless follow spot. This is done at step **808** by correlating the image that remains against a reference, to determine the proper part of the image to be converted into a shadowless follow spot. The image of the performer is separated from other things in the image. Preferably it is known for example what the performer will wear, or some image of a unique characteristic of the performer has been made. That unique characteristic is correlated against the performer image to determine the performer only at the output of step **808**. This image is digitized at step **810**: that is all parts of this image which are not performer are set to zeros so that light at those positions is reflected. In this way, a gobo-like image is obtained at step **810**, that gobo-like image being a changing cutout image of the performer. An optional step **812** further processes this image to remove artifacts, and preferably to shrink the image slightly so that it does not come too close to the edge of the performer's outline. This image is then transferred to the VRAM at step **814**, at which time it is re-entered into the DMD **1012** to form a gobo-like mask for the lamp. This allows the light to be appropriately shaped to agree with the outline of the performer **1004**.

Another embodiment of the present invention uses the above described techniques and basic system of the present invention to provide color to the lamp gobo. This is done using techniques that were postulated in the early days of color tv, and which now find a renewed use. This system allows colored gobos, and more generally, allows any video image to be displayed.

FIG. **9a** shows the lamp **310** in a series with a rotating multicolored disk **902**. FIG. **9b** shows the three sectors of the disk. Red sector **950**, a blue sector **952**, and a green sector **954**. The light along the optical path **902** is colored by passing through one of these three quadrants, and then through DMD **320**. DMD **320** is driven by a rotating source **910**, synchronized with the operation of spinning of the color disk **902**. The video is driven to produce a red frame, then a green frame, then a blue frame, one after another, for example. The red filtered video is transferred at the same moment when the red sector **950** is in the light path. So as long as the different colors are switched faster than the eye's persistence of vision, the eye will average them together to see a full color scene.

Although only a few embodiments have been described in detail above, those having ordinary skill in the art will certainly understand that many modifications are possible in the preferred embodiment without departing from the teachings thereof.

All such modifications are intended to be encompassed within the following claims.

For example, any direction deflecting device could be used in place of the DMD. A custom micro mirror device would be transparent, and have thin mirrors that "stowed" at 90° to the light beam to allow the beam to pass, and turned off by moving to a reflecting position to scatter select pixels of the light beam. The color changing devices could be any device including dichroics.

What is claimed is:

1. A method of shaping a light beam in an electronic device, comprising:

projecting a beam of light;

obtaining an electronic image indicating a desired shape of said beam of light;

using software to control an electronically-controllable shape altering element which is located in a path of the beam of light based on said image; and

projecting an image which has been shaped based on the controlled shape altering element.

2. A method as in claim **1**, wherein said control comprises processing edges of the image to change edge effects of the shape of the beam of light.

3. A method as in claim **1**, further comprising rotating said image to rotate said shape.

4. A method as in claim **1**, further comprising changing a duty cycle of the image.

5. A method as in claim **1**, wherein said projecting an image comprises obtaining an image, using an outer perimeter of said image as a stencil, projecting parts inside said stencil, and not projecting parts outside said stencil.

6. A method as in claim **1**, further comprising storing a plurality of images in a memory, selecting one of said images from said memory, and using said image to shape the beam.

7. A method as in claim **6**, wherein said image in a memory is a bit mapped image.

8. A method as in claim **7**, wherein said bit mapped image is stored in a compressed form.

9. A method as in claim **6**, wherein said image in the memory represents a vectorized version of the image.

10. A method as in claim **6**, further comprising electronically processing the image in real time while displaying the image.

11. A method as in claim **1**, further comprising using image processing software to electronically process the image in real time while displaying the image.

12. A method as in claim **11**, wherein said electronically processing comprises morphing the image.

13. A method as in claim 11, wherein said electronically processing comprises rotating the image.

14. A method as in claim 11, wherein said electronically processing comprises scaling the image to a different size.

15. A method as in claim 11, wherein said electronically processing comprises changing a look of an edge of the image.

16. A method as in claim 1, further comprising selecting a change to the image between one of edge processing, image processing, and duty cycle processing, and carrying out said operation to the image.

17. A method as in claim 1, wherein said electronically-controllable shape altering element is a digital mirror device.

18. A method of projecting a shaped light beam, comprising:

obtaining an image to use in shaping a light beam;

using said image to shape a light beam being projected;

requesting a change to said image;

dividing said change to said image into slices including at least first and second slices;

determining a time for an increment between said first slice and said second slice of changing the image; and

changing said image from a first characteristic to a second characteristic, using one of said slices at each of said times.

19. A method as in claim 18, wherein said changing said image comprises rotating said image, and wherein the rotation of the image is sliced into an amount of rotation which occurs at each time.

20. A method as in claim 18, wherein the changing the image is a shifting of position of the image, and wherein said slice is a number of pixels of shift per time period.

21. A method as in claim 18, wherein said changing comprises calculating a velocity of movement of said image, determining a number of pixels per slice for movement of said image, and determining times between slices using said pixels per slice and velocity of movement.

22. A method as in claim 21, further comprising, at each time, calculating a new image.

* * * * *

EXHIBIT K

US006326741B1

(12) **United States Patent**
Hunt et al.

(10) **Patent No.: US 6,326,741 B1**
(45) **Date of Patent: *Dec. 4, 2001**

(54) **STAGE LIGHTING LAMP UNIT AND STAGE LIGHTING SYSTEM INCLUDING SUCH UNIT**

(75) Inventors: **Mark A. Hunt**, Derby; **Keith J. Owen**, Moseley; **Michael D. Hughes**, Wolverhampton, all of (GB)

(73) Assignee: **Light & Sound Design Ltd.** (GB)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

This patent is subject to a terminal disclaimer.

(21) Appl. No.: **09/313,418**

(22) Filed: **May 17, 1999**

Related U.S. Application Data

(63) Continuation of application No. 08/994,036, filed on Dec. 18, 1997, now Pat. No. 5,921,659, which is a division of application No. 08/576,211, filed on Dec. 21, 1995, now Pat. No. 5,788,365, which is a continuation of application No. 08/077,877, filed on Jun. 18, 1993, now Pat. No. 5,502,627.

(51) **Int. Cl.**[7] **H05B 37/00**
(52) **U.S. Cl.** **315/316**; 315/312; 362/301; 362/233
(58) **Field of Search** 315/294, 303, 315/308, 317, 318, 319, 324, 292, 312, 316; 362/386, 319, 285, 276, 293, 301, 233

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,392,187	*	7/1983	Bornhorst	362/233
4,716,344	*	12/1987	Newell et al.	315/312
5,005,162	*	4/1991	Mitsumori et al.	369/44.12
5,010,459	*	4/1991	Taylor et al.	362/85
5,045,983		9/1991	Shields	367/293
5,414,328	*	5/1995	Hunt et al.	315/316
5,502,627	*	3/1996	Hunt et al.	362/286
5,588,021	*	12/1996	Hunt et al.	375/211
5,769,531	*	6/1998	Hunt et al.	362/233
5,788,365	*	8/1998	Hunt et al.	362/301
5,921,659	*	7/1999	Hunt et al.	362/286

FOREIGN PATENT DOCUMENTS

0 495 305	7/1992	(EP) .
2 643 134	8/1990	(FR) .
2 209 229	5/1989	(GB) .
2 231 138	11/1990	(GB) .
89 054 21	6/1989	(WO) .

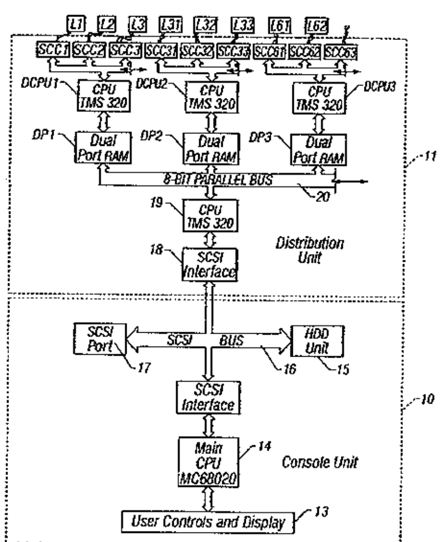
* cited by examiner

Primary Examiner—Don Wong
Assistant Examiner—Ephrem Alemu
(74) *Attorney, Agent, or Firm*—Fish & Richardson P.C.

(57) **ABSTRACT**

A stage lighting lamp unit includes a processor for receiving control data from a remote console. Beam orientation data for the lamp unit is passed to the lamp in the form of the x, y and z co-ordinates of a point in space through which the beam is to pass. The processor divides the required lamp travel into a number of stages dependent on execution duration data sent with the position data, and calculates, for each stage, a new value for pan and tilt angles for the lamp. These values are passed to pan and tilt controlling co-processors which control servo-motors for pan and tilt operation. The lamp unit also incorporates a rotatable shutter for interrupting the lamp beam when required. The shutters of all the lamps in a system can be instructed from the remote console to open and close in synchronism, thereby providing a stroboscopic effect.

43 Claims, 10 Drawing Sheets



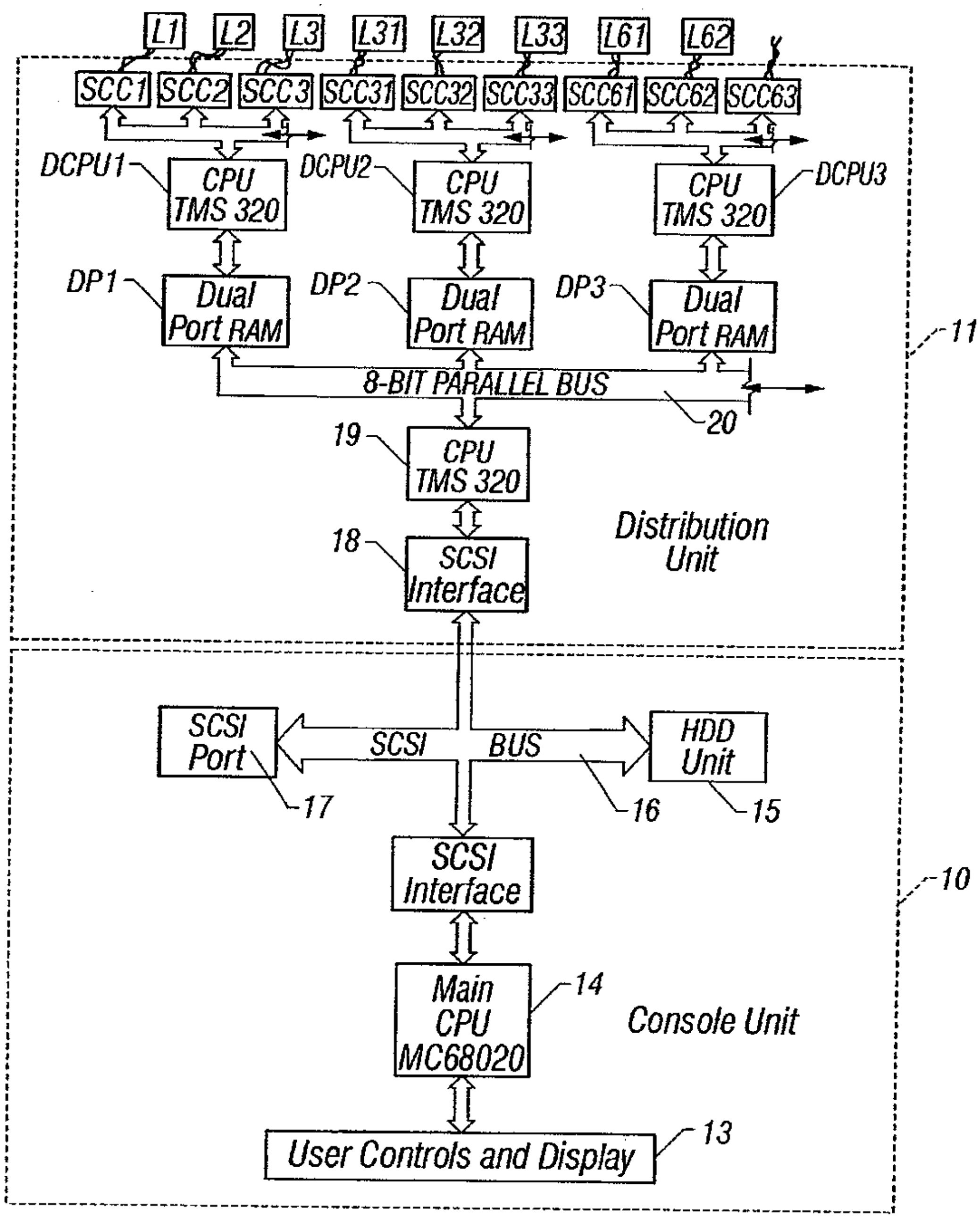

L1
L2
L3
L31
L32
L33
L61
L62
SCC1
SCC2
SCC3
SCC31
SCC32
SCC33
SCC61
SCC62
SCC63
DCPU1
DCPU2
DCPU3
CPU TMS 320
CPU TMS 320
CPU TMS 320
DP1
DP2
DP3
Dual Port RAM
Dual Port RAM
Dual Port RAM
8-BIT PARALLEL BUS
20
11
19
CPU TMS 320
18
SCSI Interface
Distribution Unit
SCSI Port
SCSI BUS
HDD Unit
17
16
15
SCSI Interface
10
Main CPU MC68020
14
Console Unit
User Controls and Display
13

FIG. 1

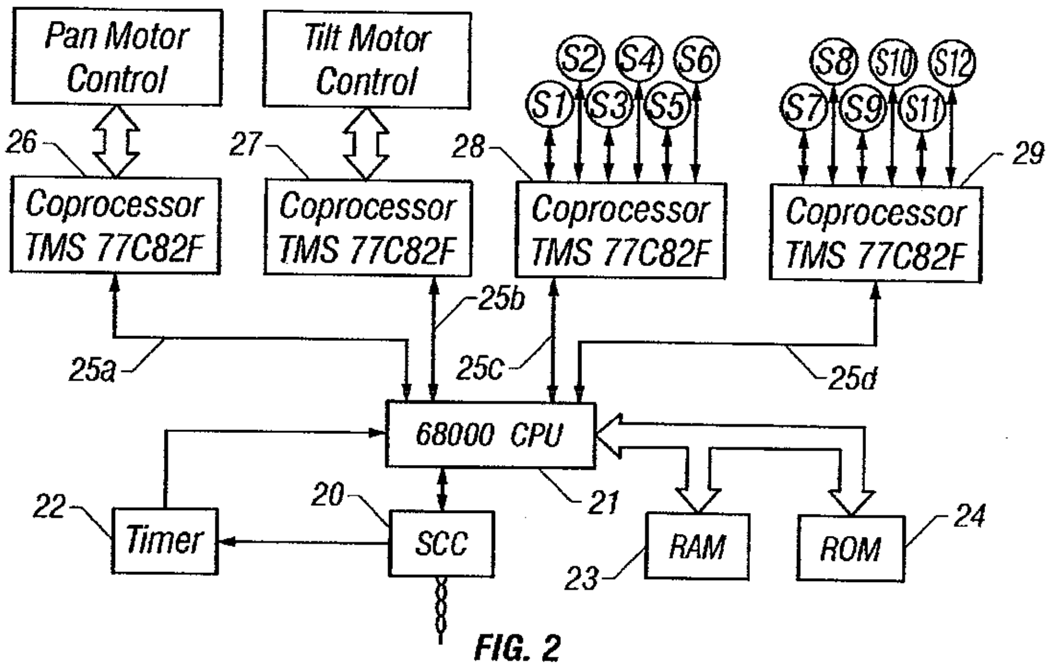

Pan Motor Control
Tilt Motor Control
S1 S2 S3 S4 S5 S6
S7 S8 S9 S10 S11 S12
26
27
28
29
Coprocessor TMS 77C82F
Coprocessor TMS 77C82F
Coprocessor TMS 77C82F
Coprocessor TMS 77C82F
25a
25b
25c
25d
68000 CPU
21
22
Timer
20
SCC
RAM
23
ROM
24

FIG. 2

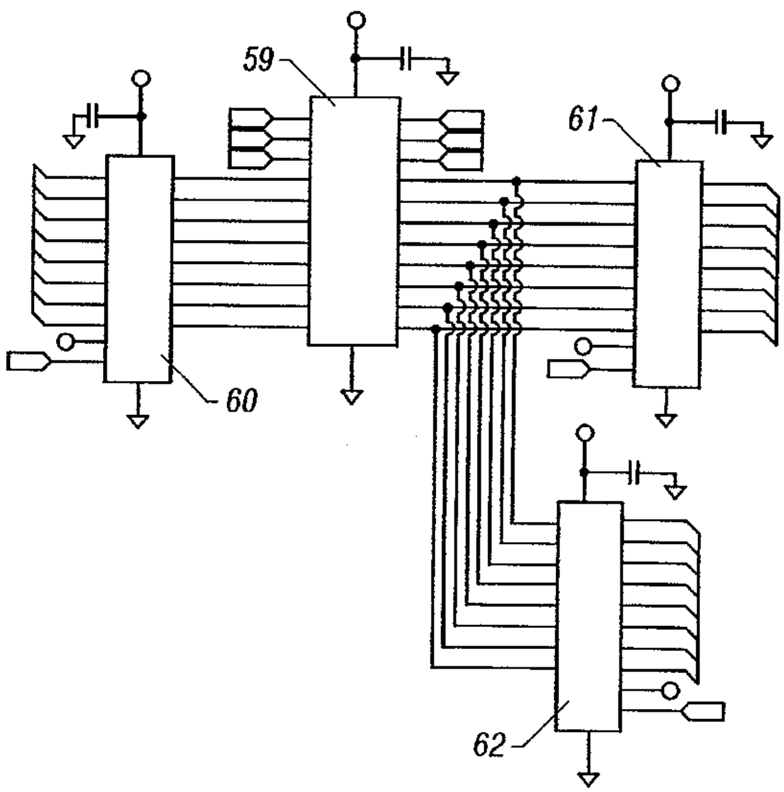

59
60
61
62

FIG. 7

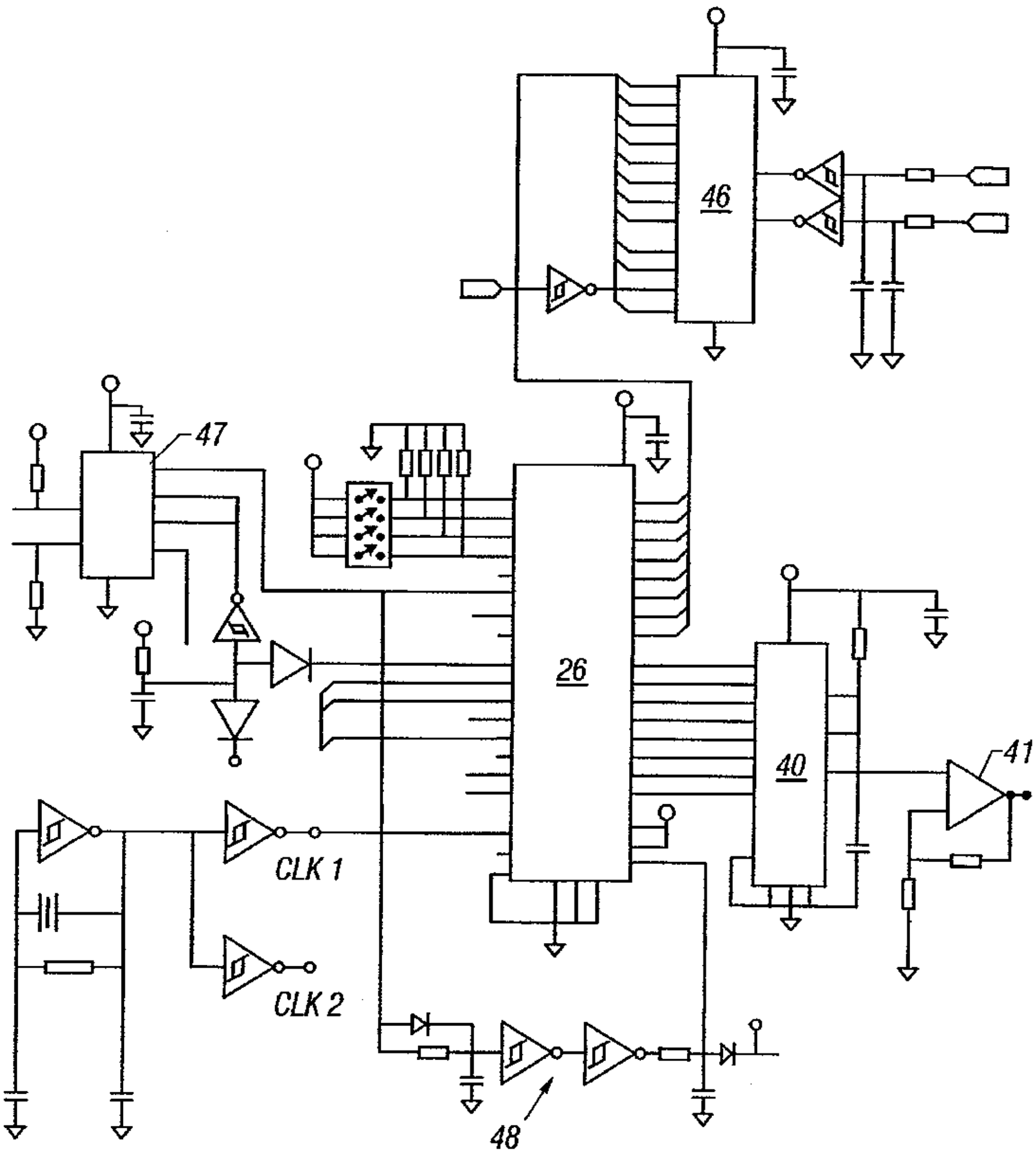


FIG. 3

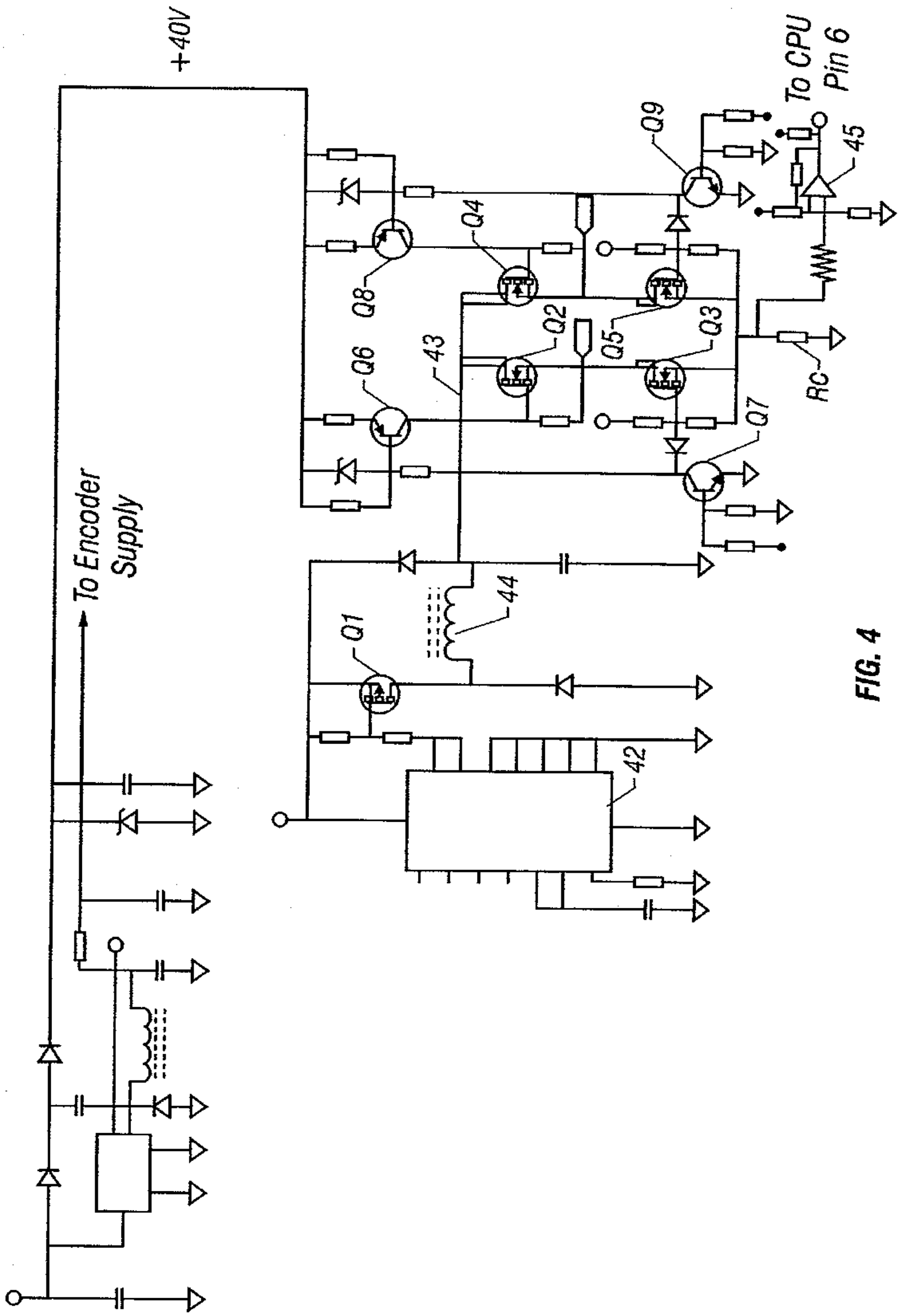

+40V
To CPU
Pin 6
Q9
45
Q4
Q8
Q2
Q5
Q3
Q6
43
Rc
Q7
To Encoder
Supply
44
Q1
42

FIG. 4

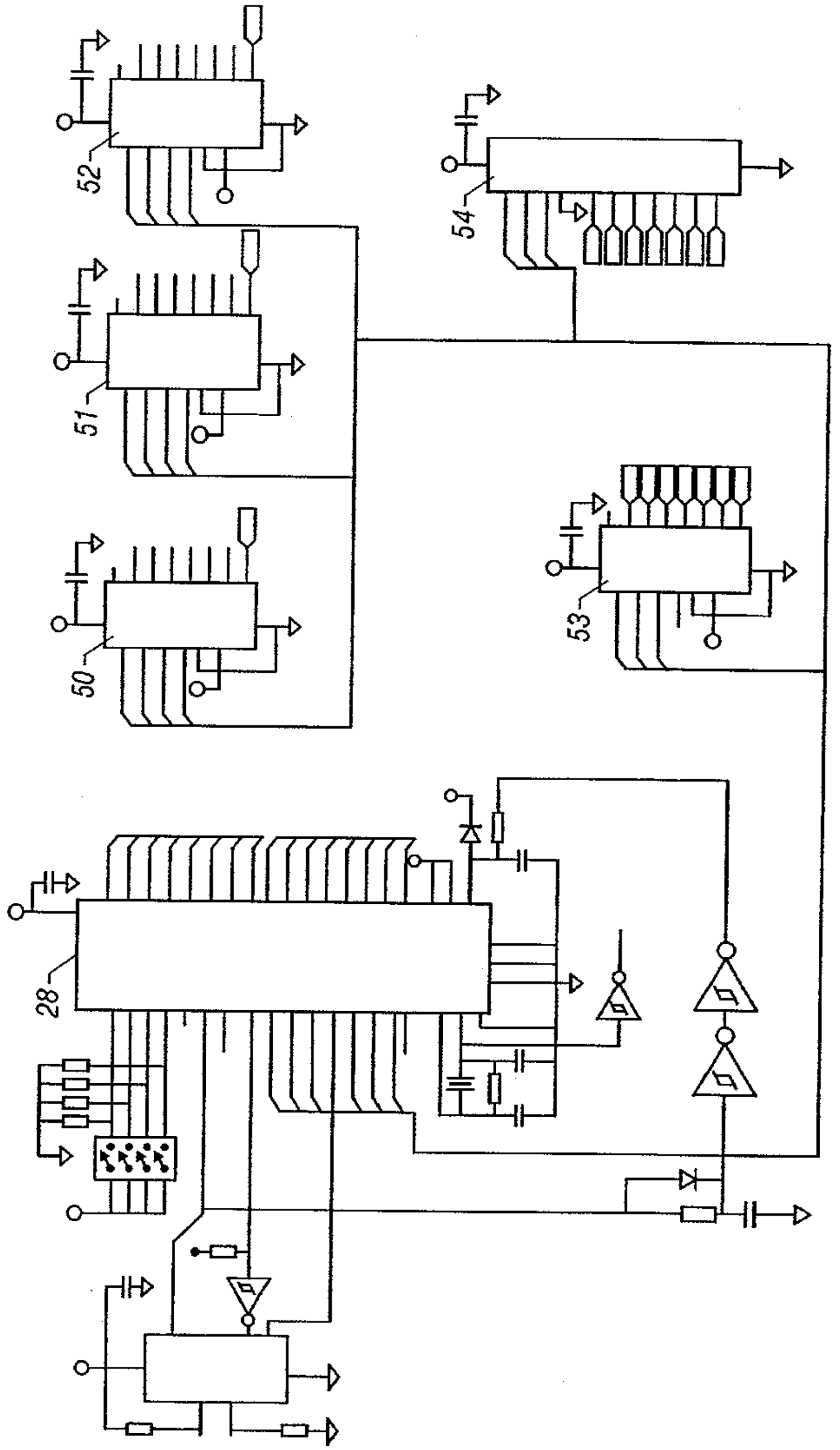


FIG. 5

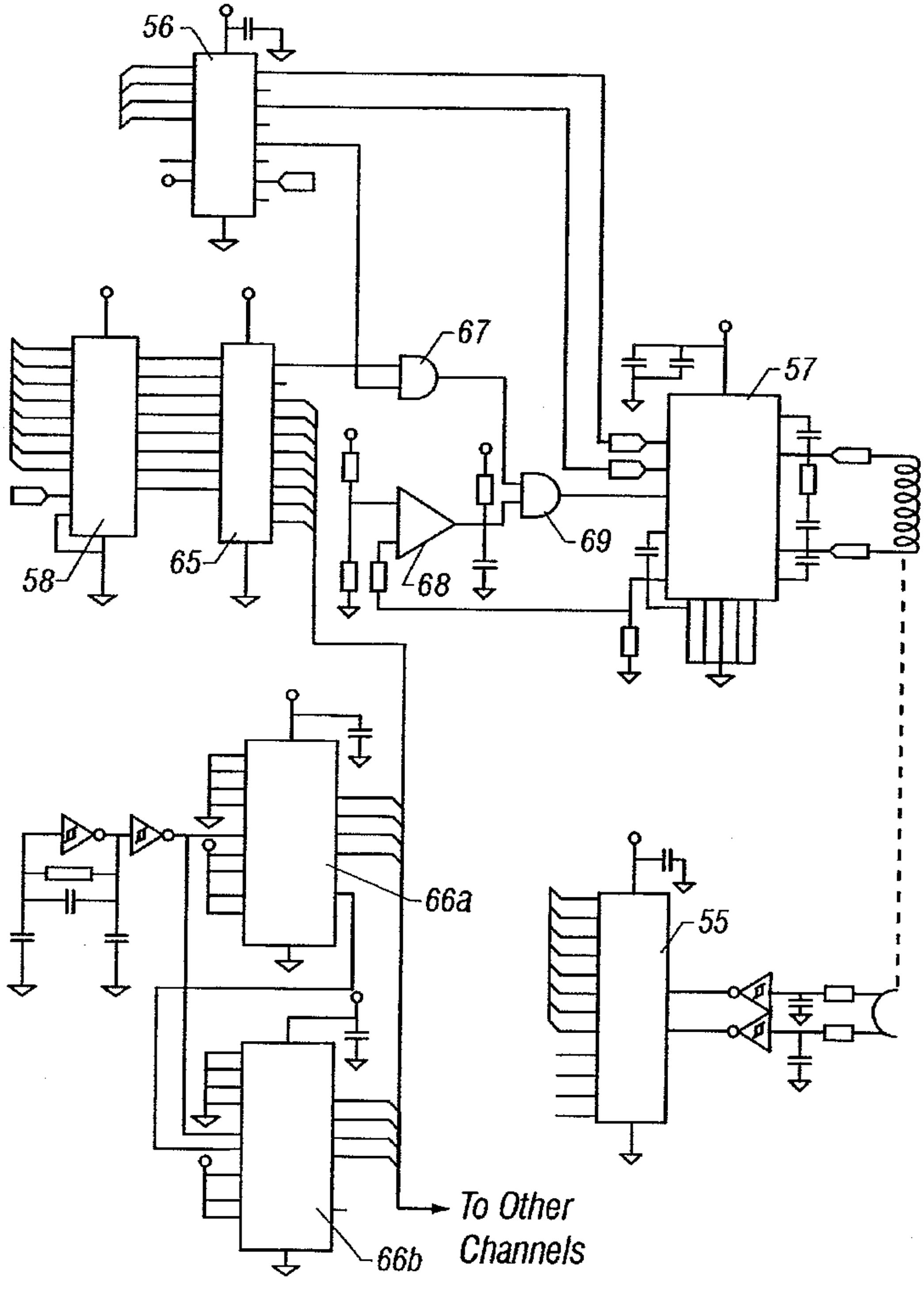


FIG. 6

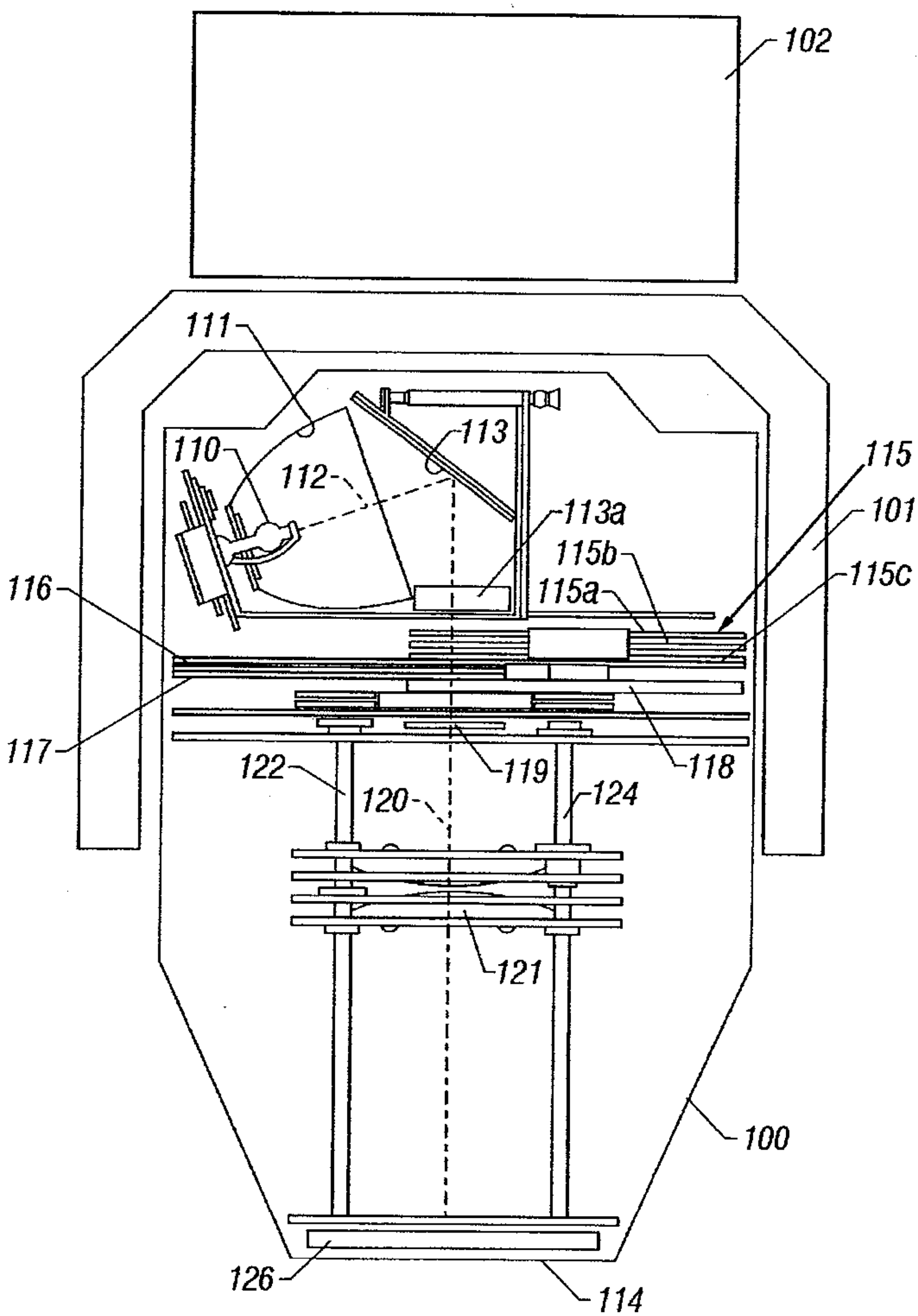

102
111
110
112
113
113a
115
101
115b
115c
115a
116
117
122
120
119
118
124
121
100
126
114

FIG. 8

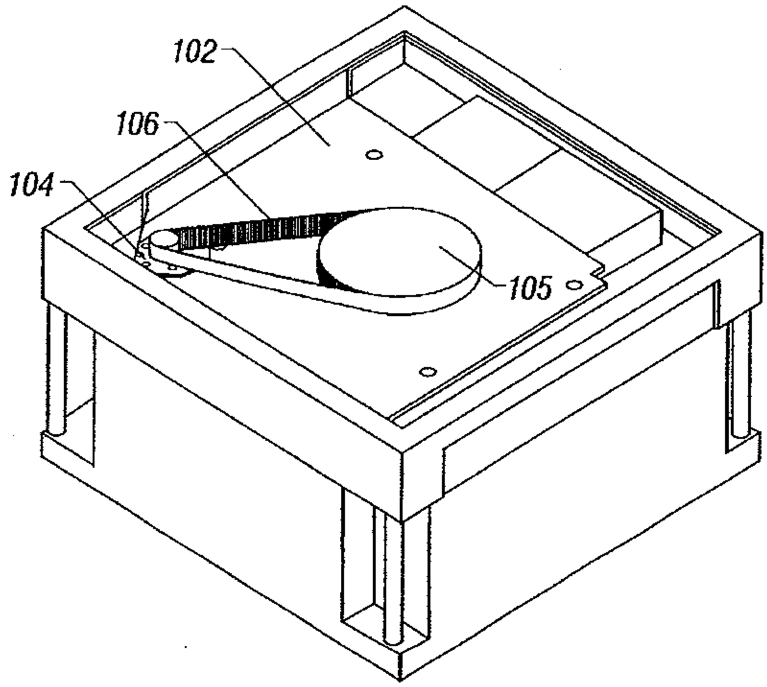


FIG. 9

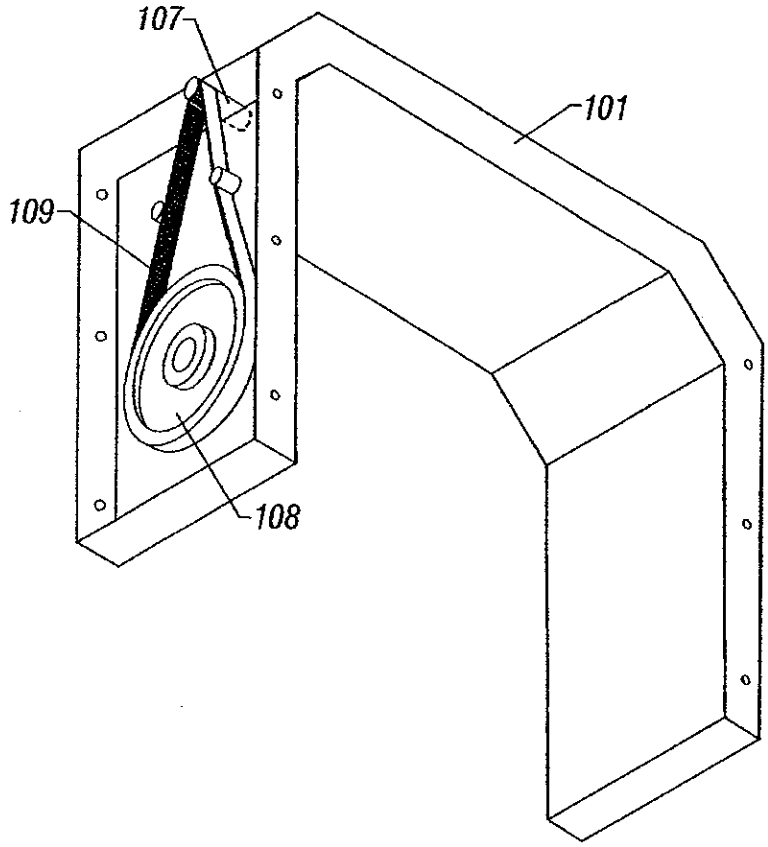


FIG. 10

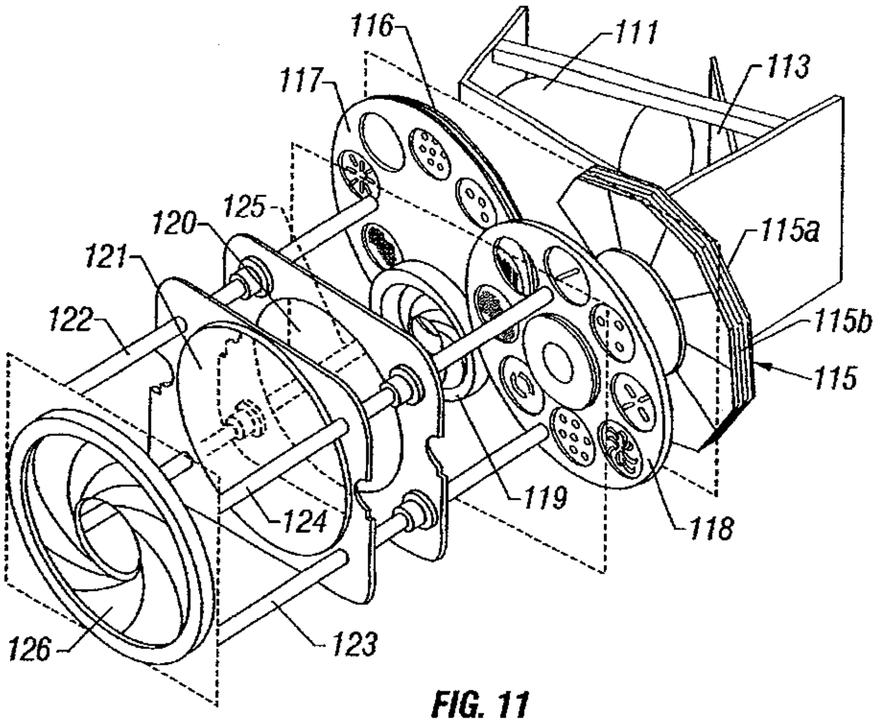

116
111
117
113
125
120
115a
121
122
115b
115
119
124
118
126
123

FIG. 11

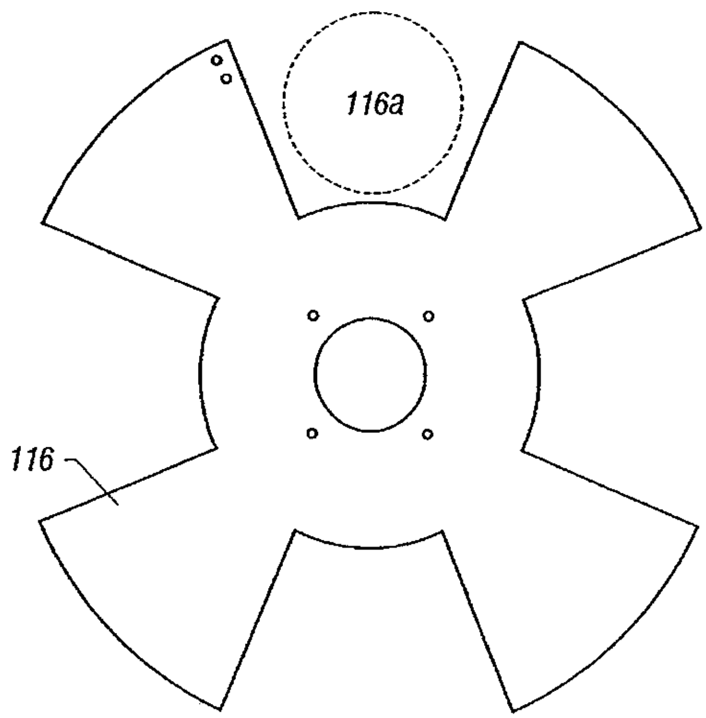

116a
116

FIG. 13

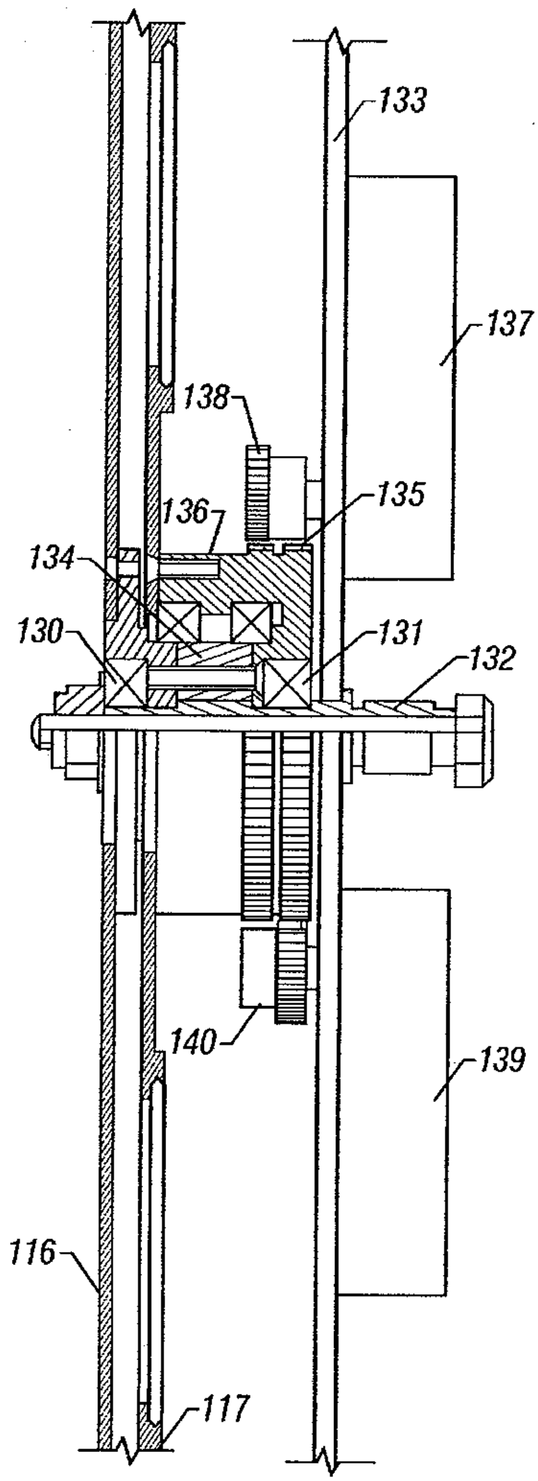


FIG. 12

STAGE LIGHTING LAMP UNIT AND STAGE LIGHTING SYSTEM INCLUDING SUCH UNIT

This is a continuation of application Ser. No. 08/994,036, filed Dec. 18, 1997, now U.S. Pat. No. 5,921,659 issued Jul. 13, 1999 which is a divisional of application Ser. No. 08/576,211, filed Dec. 21, 1995, now U.S. Pat. No 5,788, 365, which is a continuation of application Ser. No 08/077, 877 filed Jun. 18, 1993, now U.S. Pat. No. 5,502,627.

This invention relates to stage lighting and is particularly concerned with the control of multiple functions of a lamp.

It has already been proposed to incorporate in a lamp unit a plurality of different functions, such as colour changers, focusing lenses, iris diaphragms, gobo selectors and pan and tilt mechanisms which are controlled from a remote console. Stage lighting systems have as a result reached very high levels of complexity requiring a very complicated main control console and lamp unit constructions. The use of microprocessors, both in the console and the lamps has become conventional as increasing complexity makes it more difficult to produce and subsequently maintain a system which uses hard wired logic or analog controls. In such systems the microprocessor in the console is used to allow the user to set up lighting cues and to control the sending of appropriate data to the lamp microprocessors. The lamp microprocessors are also involved in controlling communication between the console and the lamps, and also have to control a plurality of servo-motors which drive the various functions of the lamps.

It is one object of the present invention to provide a lamp microprocessor and servo-control arrangement which allows complex functions to be carried out.

It is another object of the invention to provide a lamp control system in which control of pan and tilt movements of each lamp can be carried out in rapid and efficient manner, enabling large groups of lamps to make coordinated movements.

It is yet another object of the invention to provide each lamp in a stage lighting system with a means for quickly interrupting its light beam and quickly re-establishing the beam so that a group of lamps can be made, when required to flash in synchronism.

In accordance with one aspect of the invention there is provided a lamp unit for connection to a remote control console for the control of a plurality of different functions of the lamp, said unit comprising a main processor circuit, associated with a communication controller for accepting message data from the console, a plurality of servo-controls for operating said functions of the lamp, and a plurality of co-processors which are connected to the main processor circuit so as to be supplied thereby with desired value data for the various lamp functions, said servo-controls being controlled by said co-processors.

In the case of pan and tilt controls where close control is required throughout the movement of the lamp from an initial position to a new position, one of the co-processors is assigned solely to the control of movement about each axis. Other functions can share a co-processor.

The main processor circuit of the lamp is preferably programmed to accept data from the control console defining not only a target position for any function, but also a duration over which the function is to be executed. In this case the main processor circuit divides the "journey" into segments and updates the target position data passed to the associated co-processor at intervals.

In accordance with another aspect of the invention, there is provided a lighting control apparatus comprising the combination of a main control console for accepting user input relating to required beam movements, a plurality of independently operable lamp units situated remotely from the console, each of the lamp units incorporating a servo-mechanism for automatically moving the lamp beam about two mutually transverse axes to a desired angular position and data communication means connecting the console to the lamp units for the transmission of desired position data to the lamp units, the desired position data being transmitted in the form of a set of three dimensional linear co-ordinates defining a point in space through which the lamp beam is required to pass, and each lamp unit including a calculating device for calculating the desired angular position from the desired position data and supplying the servo-mechanism with such desired angular position.

In addition to the "point at" mode of operation mentioned above, additional modes may be specified in which the lamps point away from the specified point or in which they all point in the same direction parallel to a line between a fixed position in the co-ordinate system and the specified point.

Conveniently, all the data concerning the positions and orientations of the individual lamp units within the co-ordinate system is stored in a setup file kept on a hard disk drive in the console. When the same lighting set-up is used at different venues, where it is impossible to set the frame which carries all the lamp units at exactly the same position as that for which the set-up was designed, offset data can be input at the console and either used within the console microcomputer to correct the position data stored during set-up as it is sent out, or such data can be sent to all of the lamp units over the network and stored there, to enable the corrections to be made in the individual lamp processor units.

In accordance with another aspect of the invention, a stage lighting unit comprises a housing, a light source within said housing, an optical system for forming light from said light source into a beam, a rotary shutter device having a plurality of blades, said shutter device being rotatably mounted in the housing so as to cause said blades to pass through and obstruct said beam as the shutter device rotates, a motor for rotating said shutter device and a servo-control for controlling said motor in accordance with data received in use from a remote control console.

The invention also resides in a stage lighting system incorporating a plurality of lighting units as defined above controlled by a common remote control console via data communication means, whereby the rotary shutter devices of all the units can operate in synchronism.

An example of the invention will now be described with reference to the accompanying drawings, in which:

FIG. 1 is a block diagram of a stage lighting system;

FIG. 2 is a block diagram of the internal circuitry of one of a plurality of lamp units in the system of FIG. 1;

FIGS. 3 and 4 are more detailed circuit diagrams showing a pan motor drive control forming part of the internal circuitry of the lamp;

FIGS. 4 to 7 are detailed circuit diagrams showing a rotary shutter motor drive control forming part of the internal circuitry of the lamp;

FIG. 8 is a diagrammatic, part sectional view of one of the lamps;

FIG. 9 is a perspective view of a pan movement drive arrangement;

FIG. 10 is a perspective view of a tilt movement drive arrangement;

FIG. 11 is a diagrammatic perspective view of the internal moving parts or the lamp;

FIG. 12 is a sectional view showing the drive arrangement for a shutter and a gobo wheel forming part of the lamp; and

FIG. 13 is an elevation of a shutter wheel forming part of the lamp.

Referring firstly to FIG. 1, the system consists basically of a console unit 10, a signal distribution unit 11 and a plurality of lamps L1, L2, L3 . . . , L31, L32, L33 . . . , L61, L62 . . . individually connected by twisted pair data communication links to the distribution unit.

The console unit 10 has an array of switches, slider potentiometers, rotary digital encoders and other user actuable input devices (not shown) and a display indicated at 13. These are all connected to main console cpu 14 (an MC68020 micro-processor) which has the task of receiving inputs from the user actuable input devices and controlling the display. Both tasks are assisted by separate co-processors which directly interface with different parts of the console.

The main cpu can communicate with a hard disk drive unit 15 via a SCSI bus 16 which also connects it to the distribution unit and to an external SCSI port 17, through the intermediary of which the console can, if required be connected to a personal computer. The user controls can be used in setting up a sequence of cues in advance of a performance, the sequence being stored in a cue file on the hard disk drive unit 15. The sequence can be recalled during the performance to enable the various stored cues to be executed. Direct manual control of the lamps from the console is also possible as is manual editing of cues called up from the hard disk. The main console cpu 14 creates messages to be sent to the individual lamps, each message comprising a fixed number of bytes for each lamp. The messages contain data relating to the required lamp orientation, beam coloration, iris diaphragm diameter, gobo selection and rotation, zoom projection lens control and opening or closing of a shutter included in the lamp. A block of the RAM of the main cpu is set aside for the storage of these messages, the block being large enough to contain messages for 240 lamps, being the largest number which can be controlled via the distribution unit. Where it is required to control more than 240 lamps additional distribution units can be connected to the SCSI bus and extra main cpu RAM reserved for message storage. When any message data is changed the main cpu 14 sets a flag in the RAM block which is detected at a given point in the main cpu program loop and interpreted as a signal that the changed message data is to be transferred to the distribution unit 11.

The distribution unit 11 has a main cpu 19 which controls reception of data from the SCSI bus interface and distribution of such data to up to eight blocks of dual,port memory DP1, DP2, DP3 . . . via an eight bit data bus 20. The cpu 19 is alerted to the waiting message data when cpu 14 selects the distribution unit. The cpu 19 then supervises byte by byte transfer of the message data which it routes to the various blocks of dual port memory.

For actually sending out the message data to the lamps, there are a plurality of serial communication controllers SCC1 to SCC30, SCC31 to SCC60 etc, there being thirty serial communication controllers associated with each block of dual port memory. A further cpu DCPU1, DCPU2, etc is associated with each block of dual port memory and distributes message data transferred to the dual port memory to the individual serial communication controllers and the messages are transferred to the lamps. Each serial communication controller in the distribution unit includes a line driver which can be disabled except when data is to be transmitted. Enabling of the driver can cause a spurious

signal to be transmitted over the data link. To allow such spurious signals to be identified and ignored, a two-byte gap is left between enabling the line driver and commencing transmission of the message data for the channel in question.

This will be described in more detail herein. All asynchronous serial communication systems require framing information to synchronize the reception process. This has been typically done in the prior art using start bits and stop bits.

The present invention preferably uses FM0 coding in which the data is transmitted as one cycle of the carrier frequency for a zero or as a half cycle of the carrier frequency for a one. When the line has been idle, no waveform at all is present. When the line drivers are first enabled, an arbitrarily short pulse will usually appear on the line, due to lack of synchronization between the data signal and the enabling signal. This short data pulse could be misinterpreted as a start bit, for example and if so it would disturb later framing.

The present invention avoids any problems from this arbitrarily short pulse. To avoid this, the present invention uses a timer on the receive line, set to the time needed to receive two bytes on the serial data line. This timer is restarted whenever a byte on the data line is detected.

Each time the timer interrupt occurs, the number of bytes received is checked against the number of bytes in a valid data frame. If the number is incorrect, then the count is cleared and the message is discarded. If correct, the information is passed to the main program loop by setting a flag variable.

When the data line is first enabled, the distribution box has an internal delay of at least two byte times, which must elapse before any data will be sent. Any data received by the lamp will therefore be discarded as noise by the timer interrupt routine. After that, the real data can be safely sent down the line since the start bit of the first byte will be received correctly. When the transmission is completed, the line drivers will be disabled again.

Each of the cpus eg DCPU1, transfers data from the associated dual port RAM DP1 to the serial communication controller SCC1 to SCC30 with which it is associated one byte at a time, ie the first byte for SCC1 is transferred followed by the first byte for SCC2 and so on, each serial communication controller commencing transmission as soon as it has received its byte of data. The serial communication controllers operate to transmit data at 230.4 Kbps so that it takes about 35 μs to transmit each byte. Transfer of data from the dual port RAM DP1 to the serial communication controllers is, however, at a rate of several Mbps, so that the transmissions from all the serial communication controllers are almost simultaneous. The cpu DCPU1 is not required to monitor the transmission of data by the serial communication controller, but utilizes a software timer to commence transfer of the second byte to the serial communication controllers. This timer is started when transfer of the byte of data to the last serial communicationcontroller SCC30 has been completed and its time-out duration is slightly longer than the byte transmission time, say 40 μs. Transmission of all the messages takes about 1.5 ms out of a distribution unit main program loop duration of 4 ms.

As shown in FIG. 2, each lamp includes a serial communication controller 20 which controls reception of message data from the individual data link connecting it to the distribution unit 11. The receipt of any signal from the data link causes an interrupt of the lamp main cpu 21 (another MC68000) and the cpu 21 then controls acceptance of the signals. A timer 22 times the gaps between bytes received

from the data link and this timer causes another interrupt on time-out. The time-out time of the timer is between the times taken to transmit 1 and 2 bytes, so that time out always occurs following a spurious signal caused by line driver enabling. The time-out interrupt causes the cpu **21** to inspect the total number of bytes received since the initial interrupt and if this is less than the expected number of bytes (which is constant) the message is ignored. The time-out interrupt also resets a software data pointer to the beginning of a receive buffer in readiness for the next transmission.

The cpu **21** operates in accordance with programs stored in the lamp cpu ROM. On receipt of a message of valid length, a program variable representing the number of messages received since the lamp program was last started is incremented and the main program loop of the lamp cpu checks this variable every 16 mS. If the variable has changed since the last check, the data in the receive buffer is compared with corresponding values of variables representing current "desired values" of the various lamp function parameters. For example the receive buffer may contain two bytes representing the x, y and z co-ordinates of a point in an orthogonal three dimensional frame of reference, through which point it is required that the axis of the lamp beam should be directed. If the values of the corresponding byte pairs in the receive buffer and the desired value variables already contained in the cpu RAM are the same, no action is taken in respect of the control of the motors which control pan and tilt action of the lamp (to be described in more detail hereinafter).

As shown in FIG. **2**, the main lamp cpu **21** communicates via serial data links **25*a*, 25*b*, 25*c*** and **25*d*** with four servo-control co-processors **26, 27, 28** and **29**. Each of these co-processors is a TMS77C82 cpu. Co-processors **26** and **27** respectively control pan and tilt operation, and each of the co-processors **28** and **29** can control up to six different dc servo-motors operating different functions of the lamp.

Before proceeding with a more detailed description of the circuitry and operation of the lamp electronics, some detail will be given of the various functions of the lamp. FIG. **8** shows the relative positions of a plurality of independently operable beam characteristic control elements within the lamp housing **100**. The lamp housing is pivotally mounted on a U-bracket **101**, which is itself pivotally mounted on a mounting base **102**. FIG. **9** shows the mounting base **102** which incorporates a pan drive motor/gearbox/optical encoder arrangement **104** which drives a gear **105** attached to the U-bracket via a reduction toothed belt drive **106**. FIG. **10** shows how, within the hollow structure of the U-bracket **101**, there is mounted a tilt drive motor/gearbox/optical encoder **107** which drives a gear **108** attached to the lamp housing via another reduction toothed belt drive **109**.

As shown in FIGS. **8** and **11**, within the lamp housing, a light source **110** is mounted within an ellipsoidal reflector **111** providing a light beam with an axis **112** which is reflected by a mirror **113**, which is a dichroic mirror that reflects only visible light and passes ultra voilet and infra red light, the reflected light passing out through an opening **114** at the opposite end of the housing. The reflector **111** has a generally cup-shape surrounding the bulb **110**. According to one aspect of the invention, the axis **112** has an angle pointing in a direction rearward relative to a perpendicular to the central axis **120** of the lamp unit. If the reflector is located as shown, such that an outside edge of the reflector is generally parallel to a rear end of the housing, the optimal packing efficiency is achieved. As shown in FIG. **8**, this allows the reflector to be most efficiently packed into the available space. The reflected beam from the mirror **113** passes firstly through a collimating lens **113*a***, and then the colour changer **115** which comprises dichroic filters having differing transmission characteristics mounted on co-centered three filter disks **115*a*, 115*b*** and **115*c*** rotable around a common axis of rotation. Each disk has nine different filters on it and one blank space around its periphery, so that up to 1000 different combinations of filters can be positioned across the beam by selective positioning of the three disks (although not all of these combinations are necessarily useful as some may block all visible light). The blank space of each of the disks can be used to eliminate any color changing characteristic of that disk. These disks are driven by three of the dc servo-motors. Next the light beam passes through the plane of a bladed shutter **116** (shown in FIG. **13**) and a first gobo wheel **117** which has various gobos mounted in or over circular holes therein. As shown in FIG. **12** described in more detail hereinafter, two motors are committed to driving the shutter **116** and the gobo wheel **117** respectively. Next, there is a second gobo wheel **118** on which there are mounted a plurality of gobos which are rotatable relative to the wheel **118**. There is one motor (not shown) for driving the gobo wheel **118** and another for rotating the gobos mounted thereon through a gear arrangement (not shown). Next along the light beam is a beam size controlling iris diaphragm **119** driven by another motor (not shown). Two further motors (not shown) drive two lens elements **120, 121** along guides **122, 123** parallel to the beam axis using lead screws **124, 125**. The lens elements form a simple two element zoom lens controlling the spread and focus of the beam. Finally, an outer iris diaphragm **126** is provided adjacent the opening **114** and this is driven by a further motor (not shown). In the example described, therefore only eleven channels are actually employed.

Referring now to FIG. **12**, the shutter **116** is rotatably mounted on bearings **130, 131** on a shaft **132** fixed to a mounting panel **133** which is secured to the housing. The gobo wheel **117** is rotatably mounted on bearings on a tubular shaft **134** which acts to space the shutter **116** from a first drive gear **135**. The gobo wheel **117** is actually mounted on a second drive gear **136**. The shutter motor **137** (which is combined with a reduction gearbox and an optical encoder) is mounted on the panel **133** and drives a pinion **138** meshed with the first gear **136**. Similarly motor **139** drives a pinion **140** meshed with the second gear **136**. The shutter has four blades arranged symmetrically around its axis, with the blades and the gaps between them each subtending 45 degrees at the axis. The blades and the gaps between them are wide enough to block or clear the entire cross-section of the beam, shown in FIG. **13** at **116*a***.

Turning now to FIGS. **3** and **4**, the co-processor **26** is shown providing an eight bit data output to a d/a converter **40** (FIG. **3**) the output of which is amplified by an operational amplifier **41** and supplied to the "COMPEN" terminal of an LM3524 pulse width modulator ic **42** (FIG. **4**). The ic **42** control a P-channel enhancement mode MOSFET **Q1** which, when switched on, connects a 24V supply to a motor supply bus **43** through the intermediary of an inductor **44**. The motor is connected in a bridge formed by two push-pull pairs of MOSFETs Q2, Q3 and Q4, Q5. These four MOSFETs are driven by respective driver transistors Q6, Q7, Q8 and Q9. Transistors Q7 and Q9 are respectively controlled by "LEFT" and "RIGHT" outputs taken from the co-processor **26**, so that FETs Q2 and Q5 or FETs Q3 and Q4 are biassed to conduct. Transistors Q6 and Q8 are driven from a 40V supply rail so as to ensure that FETs Q2 and Q4 are turned hard on when conductive, thereby ensuring minimum power dissipation in these devices.

The two FETs Q3 and Q4 are connected to the return bus via a current sensing resister RC, which supplies a current related signal to a voltage comparator **45** with hysteresis to provide an input to the A6 input terminal of the co-processor **26** when the current exceeds a predetermined limit. This enables the co-processor to reduce the power applied to the motor to maintain it within safe operating limits.

The optical encoder of the pan motor provides two digital outputs in quadrature, these outputs being cleaned up by interface circuits and applied to two inputs of an HCTL-2016 counter ic **46** intended specifically for use with quadrature type encoders. The counter **46** counts up when the pulses are in one relative phase relationship and down when the opposite phase relationship exists. It therefore maintains a count-state related to the motor shaft position and hence the pan angle of the lamp. This count-state is applied to the C0 to C7 terminals of the co-processor **26**. The co-processor **26** also receives "desired value" data from the main lamp cpu **21**, via a 75176 ic **47** (which in fact serves both co-processors **26** and **27**). The Ic **47** is used to control the transmission of data between the main lamp cpu and the co-processors. Normally the ic **47** is set to receive data from the cpu **21** and pass it to the two co-processors **26** and **27**. At power-up or when the main lamp cpu **21** transmits a "break" command, the co-processor **26** is reset by a circuit **48**. The co-processor **26** has a cycle time of 1 mS and on receipt of new data it determines the distance to be travelled and then increases the "desired position" value which is compared with the actual position count by one sixteenth of the required change on each successive iteration of its control loop.

The desired value signals passed from the cpu **21** to the co-processor **26** are also time-sliced, being incremented every 16 mS. When new position data is transmitted to the lamp it is accompanied by data representing the length of time over which the movement is to be spread. The data is received, as mentioned above, in the form of two byte numbers respectively representing the x, y and z co-ordinates of a point in a Cartesian co-ordinate system. During initial setting up of the system, each lamp is sent data which informs its cpu **21** of its position in the co-ordinate system and also of its orientation.

On receipt of a new set of "point at" co-ordinates, the cpu **21** undertakes a "time-slicing" operation to determine how data should be passed to the co-processors **26** and **27**. First of all, it determines how many 16 mS loops will take place in the time duration determined by the data contained in the massage received by the lamp and sets up a variable U equal to the reciprocal of this number. A travel variable P is initialised to zero and the total distance to be travelled is determined for each of the pan and tilt movements. Thereafter, on every iteration of the 16 mS loop the travel variable P is incremented by the reciprocal variable U, the result is multiplied by the total travel required and this is added to (or subtracted from) the previous desired value before transmission to the co-processor **26** or **27**. When the variable P exceeds unity, the target has been reached.

The message sent to the lamp may include a flag indicating whether travel is to occur in a linear fashion as described above or have a sinusoidal profile imposed on it. In the latter case the value of P is modified as follows:

$P'=\sin(2*P)+0.5*(P>0.5)$ the latter term being 0 or 1

The main cpu **26** must next convert the x,y,z values into pan and tilt value data for passing to the co-processors **26** and **27**. The cpu first carries out a linear transformation of the absolute x,y,z co-ordinates into co-ordinates x',y',z' relative to the lamp's own frame of reference using the data supplied during initial set up. The ratio of the transformed x' and y' values is calculated as a 16 bit integer, which is used as an index to an ARCTAN table stored in ROM to obtain a value for the desired pan angle. To find the tilt angle, it is first necessary to establish the radial position of the target point in the transformed horizontal plane by calculating the square root of the sum of the squares of the coordinates x' and y'. In carrying out this calculation it is necessary to detect an overflow condition which exists if the sum of the squares is a 33 bit number. If this condition is detected, each square is divided by four and a new sum is formed, an overflow flag being set to indicate that overflow has occurred. The square root is found by up to sixteen steps of successive approximation and the result is doubled if the overflow flag was set during the calculation. The resulting square root is divided by the value z' and the result is applied as before to the ARCTAN table to determine the tilt angle. The results obtained represent the new pan and tilt positions to which the lamp is to be moved.

The arrangement described for sending out x, y and z co-ordinate data instead of pan and tilt angle data is highly advantageous in that it enables the console main cpu load to be significantly reduced and also makes it very easy for a console operator to control light beam movements. It is frequently required for a group of lamps to be used together to illuminate a single performer. Where the performer moves from one position on stage to another it is required for all the lamps to change position simultaneously to follow. If the system involved transmission of pan and tilt angle data, this data would be different for every lamp in the group. It would have to be set up by the console operator and stored in cue files on the hard disk drive unit **15**. This would be a very time consuming operation as the pan and tilt angles for each lamp would have to be established and recorded individually. The cue record would need to be of considerable size to record all the different data for each lamp. With the arrangement described above, however, only the x,y,z coordinate data needs to be stored and when the cue is recalled the same data is sent to each of the lamps in the group.

Whilst it is theoretically possible to use stored cue data in x,y,z co-ordinate form and to use the console main cpu **14** to calculate the pan and tilt angles to send to the lamps, this would be unsatisfactory as the calculations involved would impose a very heavy load on the cpu **14**, particularly where a large number of lamps in several different groups had to be moved as the result of a single cue.

As described above a "point-at" mode is envisaged as the normal operating mode. However, other modes of operation are also envisaged. For example, the lamp could be instructed to point away from the point specified or to point in a direction parallel to a line joining a fixed point (eg the origin of the co-ordinate system) to the point specified. These "point-away" and "point parallel" modes would be selected by means of flags included in the data transmitted to the lamps.

The arrangement described enables the lamps to be very precisely synchronised. The data is transmitted from the distribution unit to all of the lamps simultaneously and each lamp can start to respond at the end of the message. This enables very precise direction of all the lamps to a moving point in "point-at" mode and very clean parallel sweeps to be made in "point parallel" mode.

It should be noted that the use of x,y,z co-ordinates is also very advantageous in situations where a pre-arranged lighting performance is to be used in several different venues. The pre-loaded gantries or trusses used for such touring performances cannot always be mounted at exactly the

required positions relative to the stage because of local conditions. In this case all that is needed is for offsets data to be sent to the lamps at set-up time to enable each lamp cpu to correct its position data. No editing of the individual pre-recorded cues is necessary as it would be in the same circumstances if pan and tilt data were stored.

As part of the set-up procedure for each performance it is necessary to initialise the values of the actual pan and tilt angle count-states, since encoders of the type used do not give any absolute position data. This is accomplished by driving the lamp to an end stop in one direction for each movement. The lamp is driven back to a predetermined number of counts and the counters are reset to zero at this position.

Turning now to FIGS. **5** to **7**, the circuitry for controlling the individual dc servo-motors inside the lamp is more complex as each co-processor has to deal with up to six servo-motors. As shown in FIG. **5**, the co-processor **28** controls a number of data routers **50** to **54** which determine which channel is being controlled at any given time. The router **50** co-operates with six HCTL-2016 counters **55** which count the quadrature pulse outputs of the respective encoders, to determine which of the counters should supply its count-state to the co-processor **28**. Router **51** controls individual resetting of the counters **55**. Router **52** co-operates with a 74HC175 ic **56** (one for each channel) to determine which L6202 ic motor controller **57** is enabled and also routes "RIGHT" and "LEFT" signals from the co-processor to the circuits **57**. Router **53** controls routing of position error data calculated by the co-processor **28** for each channel to latches **58** (one for each channel) at the input of pulse width modulator circuits for controlling the motor controllers **57**. This error data is actually passed to the latch **58** in an inverted form, so that the larger the error, the smaller the value passed is. Router **54** routes various digital sensor signals to a sensor input of the co-processor, Such sensors are utilized by some of the channels to indicate when the moving part in question is in a datum position. This is required for the gobo wheels, the colour wheels and the shutter, but not for the iris diaphragms or lenses which can be moved to end stop positions. During datum set-up the sensors (optical sensors sensing a hole or flag or Hall effect sensors) are detected and the HCTL counters are reset.

As co-processor **28** has only 256 bytes of internal memory, extra memory is required for each channel to store program variables. The RAM selection control circuit is shown in FIG. **7**. The memory ic **59** (an HM6116LP ic) has 11 address lines of which eight are connected to the co-processor write bus via a latch circuit **60** and the remaining three of which are connected to spoare outputs of three of the ics **56**. Spare outputs of the selectors **50, 51, 52** are connected to control terminals of the memory ic and a spare output of the selector **53** is connected to an output enable terminal of the latch circuit **59**. Thus a particular address in the memory ic can be selected by the co-processor by first setting the ics **56** and the selectors **50, 51, 52** to appropriate states and then outputting the lower bytes of the address to latch **60** whilst output from latch **60** is enabled. Two further eight bit latches **61** and **62** provide temporary storage for data to be written to and data just read from the memory ic **59**. When neither reads nor writes are required the memory data bus is tri-stated. Bus contention is thus avoided.

Circuit **57** actually controls the motor current, but it in turn is controlled by a pulse width modulator circuit, comprising the latch **58** and a digital comparator **65** which compares the contents of latch **58** with the count-state of an 8-bit continuously running counter **66***a*, **66***b* serving all channels. The comparator output goes high when the count-state exceeds the latch contents, so that if the latch content is low the comparator output is high for a high proportion of each cycle of the counter **66***a*, **66***b*. The output of the comparator **65** is ANDed with an enable output from ic **56** by a gate **67** and then with the output of an overcurrent detector circuit **68** by another gate **69**.

When a new target value for one of the parameters controlled by co-processor **58** arrives in the receive buffer, and it is associated with execution duration data (this may apply to lens movements, colour changer movements, gobo movements and iris diaphragm movements, but not shutter movements) the cpu **21** handles time slicing as in the pan and tilt operations. Since several channels are controlled by each co-processor, however, no interpolation by the co-processor is used. Instead each channel has its error checked and a new value written (if necessary) to latch **58** every 12 mS

In the case of the shutter, the message received by the lamp merely includes a shutter open or shutter closed command. When the required shutter status changes, the main cpu merely increases the target shutter angle by 45 degrees (in the case of a four bladed shutter) and passes the new value to the co-processor.

This arrangement enables the shutters of some or all of the lamps to be operated in synchronism. Moreover, the console cpu **14**, can operate to update the shutter open/closed instructions at regular intervals to obtain a stroboscopic effect, synchronised for all the lights.

What is claimed is:

1. A computer controlled moving light, comprising:

a light source, with a reflector assembly that is associated with the light source, said reflector having at least one rounded cross section, said light source operating to producing a light beam along an axis;

a housing having an interior shape defining an optical axis through a center portion thereof;

a beam reflector, located in said housing to reflect the light beam at least partly along said optical axis,

said reflector assembly having an inner surface which reflects the light beam and an outer surface, and defining a reflector optical axis which in non-parallel to said optical axis to emit the optical beam at an angle which forms a non zero angle with said optical axis.

2. The moving light as in claim **1** further comprising a plurality of lenses and color changing filters.

3. The moving light as in claim **2** further comprising a multiposition shutter, having a first rotational position which totally blocks a light beam from passing along said optical axis and a second position which allows the light beam to pass along the optical axis.

4. The moving light as in claim **1** wherein said reflector assembly is an ellipsoidal reflector and said outer surface of said reflector assembly is parallel to a portion of said inner surface that is non parallel with said reflector optical axis.

5. The moving light as in claim **4** further comprising an optical encoder, monitoring a position of said light source.

6. The light as in claim **4** wherein there are two quadrature optical encoders, respectively monitoring said pan movement of said light source and tilt movement of said light source, and further comprising a counter, monitoring outputs of said optical encoder, said counter counting up when pulses from said two quadrature encoders are in a specified phase relation and counting down when said pulses are in an opposite phase relation to said specified relation.

7. The moving light as in claim **6** further comprising a coprocessor which compares value data indicative of a

desired pan or tilt position with actual position of pan or tilt as detected from said counter.

8. A lighting system, comprising:

a control console, having controls for a plurality of lights thereon; and

a plurality of lights, all coupled to the control console, said control console preparing a first command relative to a first coordinate system, and sending said first command to at least first and second lights which are physically located in different positions, said first light including a processing element which converts information in said first command into a second coordinate system, and the second light including a processing element which converts said information into a third coordinate system different than said second coordinate system, each of said lights being movable, and said first and second lights moving to locations based on said information in said second and third coordinate system respectively.

9. The system as in claim 8 wherein said first coordinate system is an x, y, z coordinate system, said second coordinate system is a local coordinate system of said first light relative to a target, and said third coordinate system is a coordinate system of said second light relative to a target.

10. The system as in claim 8 wherein said lights each point at a location represented by the information in said first coordinate system.

11. The system as in claim 8 wherein said lights each point away from a location represented by the information in said first coordinate system.

12. The system as in claim 8 wherein said lights each point parallel to a location represented by the information in said first coordinate system.

13. The system as in claim 8 wherein said lights each include encoders which produce pulses indicating movement of said light into orthogonal directions, said pulses being counted to determine an absolute position of said light.

14. The system as in claim 13 where one of said encoders is used to determine a pan direction and the other of said encoders is used to determine a tilt direction.

15. The system as in claim 8 wherein said console also sends a flag to each of said lights, said flag indicating a use of the coordinates in the first coordinate system.

16. The system as in claim 15 wherein the flag indicates whether the information indicates a target to point at, or some operation other than point at.

17. The system as in claim 15 wherein the flag indicates whether the information represents a target to point parallel, or some operation other than point parallel.

18. The system as in claim 15 wherein the flag indicates whether the information represents a target to point away from or some operation other than point away.

19. The system as in claim 15 wherein the flag indicates whether the information represents point at, point parallel or point away information.

20. The system as in claim 8 wherein said information in said first coordinate system is absolute information.

21. The system as in claim 8 wherein said information in said first coordinate system is offset information.

22. A method of controlling a plurality of moving lights, comprising:

producing a message indicating a target in a first coordinate system;

simultaneously transmitting said message to a plurality of lamps at different locations; and

responding, in each of said lamps, to an end of said message by moving substantially simultaneously to point relative to said target.

23. The method as in claim 22 wherein each of the lamps point at the target.

24. The method as in claim 22 where each of the lamps respond to point parallel to the target.

25. The method as in claim 22 wherein each of the lamps respond to point away from the target.

26. The method as in claim 22 further comprising, in each of the lamps, converting the information from said first coordinate system to the lamp's own internal coordinate system which is dependent on the lamp's position.

27. The method as in claim 22 wherein the first coordinate system is an x, y, z coordinate system.

28. The method as in claim 26 wherein the lamp's own internal coordinate system is based on pan and tilt from a specified reference point.

29. The method as in claim 22 further comprising calibrating each of the lamps by initiating actual pan and tilt angle count states.

30. The method as in claim 22 further comprising calibrating the lamp by counting pulses from a encoder, to determine a distance from an initial count state in the lamp.

31. The system as in claim 22 wherein said information in said first coordinate system is absolute target data.

32. The method as in claim 22 wherein said information in said first coordinate system is offset data.

33. The method as in claim 22 wherein each lamp includes a processing element which converts information from said first coordinate system to its own internal coordinate system based on a position of the lamp.

34. The method as in claim 22 further comprising sending execution duration data from said console to the lamp.

35. The method as in claim 34 further comprising operating the lamps based on said execution duration data to allow said lamps to operate in synchronism.

36. A lighting system, comprising:

a console, producing outputs for a plurality of movable lamps, said outputs including information in a first coordinate system indicating a target where said lamp will point, targets for at least two of said lamps being the same, and the same first coordinate system information being sent both of said two of said lamps.

37. The system as in claim 36 further comprising obtaining and sending operation duration information to said two of said lamps, to allow said two of said lamps to operate in synchronism.

38. A method of controlling a plurality of spaced apart moving lights, comprising:

determining a target for said moving lights;

sending the same indication, indicative of said target, to both of said lamps; and

in said lamps translating said indication, to information indicative of a relationship between each said lamp and said target and using said information to move said lamp.

39. The method as in claim 38 wherein said lamp is moved to point at said target.

40. The method as in claim 38 wherein said lamp is moved to point away from said target.

41. The method as in claim 38 wherein said lamp is moved to point parallel to said target.

42. The method as in claim 38 wherein said lamps are moved in synchronism.

43. The method as in claim 38 further comprising sending execution duration data to the lamp which control a length of the time it takes the lamps to move positions.

* * * * *

EXHIBIT L

US006466357B2

(12) **United States Patent**
Hunt

(10) **Patent No.: US 6,466,357 B2**
(45) **Date of Patent: Oct. 15, 2002**

(54) **PIXEL BASED GOBO RECORD CONTROL FORMAT**

(75) Inventor: **Mark Hunt**, Derby (GB)

(73) Assignee: **Light and Sound Design, Ltd.**, Birmingham (GB)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/882,755**

(22) Filed: **Jun. 15, 2001**

(65) **Prior Publication Data**

US 2001/0050800 A1 Dec. 13, 2001

Related U.S. Application Data

(63) Continuation of application No. 09/500,393, filed on Feb. 8, 2000, now Pat. No. 6,256,136, which is a continuation of application No. 09/145,314, filed on Aug. 31, 1998, now Pat. No. 6,057,958.

(60) Provisional application No. 60/065,133, filed on Nov. 12, 1997, and provisional application No. 60/059,161, filed on Sep. 17, 1997.

(51) **Int. Cl.**[7] **G02B 26/00**; G06K 9/00; H04N 1/46

(52) **U.S. Cl.** **359/291**; 382/181; 382/203; 382/228; 358/537; 358/540; 345/418; 345/431; 348/26

(58) **Field of Search** 359/291; 345/127, 345/418, 431; 382/162, 181, 190, 192, 194, 195, 197, 198, 199, 200, 203; 348/110, 26, 27; 358/537, 538, 540

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,828,485	A		10/1998	Hewlett	359/291
5,852,443	A		12/1998	Kenworthy	345/431
5,878,413	A	*	3/1999	Agrawal et al.	382/181
5,995,660	A	*	11/1999	Andoh et al.	382/181
6,091,852	A	*	7/2000	Entleitner et al.	382/181
6,185,330	B1	*	2/2001	Hirao	382/181
6,256,136	B1	*	7/2001	Hunt	359/291
6,272,241	B1	*	8/2001	Tattersall	382/181
6,285,783	B1	*	9/2001	Isomura et al.	382/181
6,317,517	B1	*	11/2001	Lu	382/228

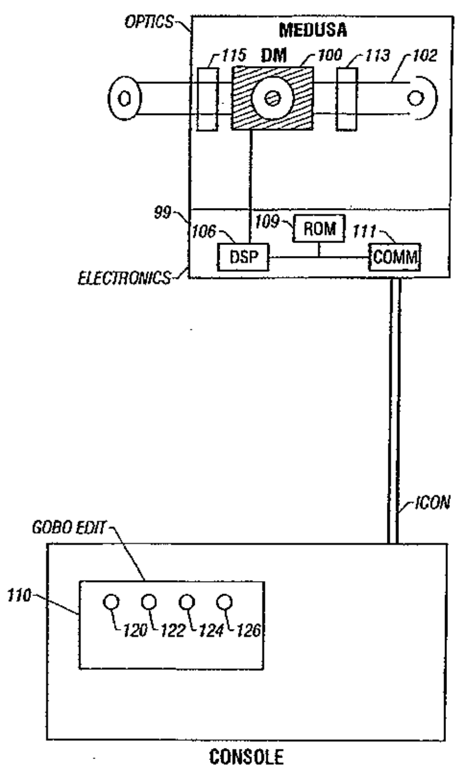
* cited by examiner

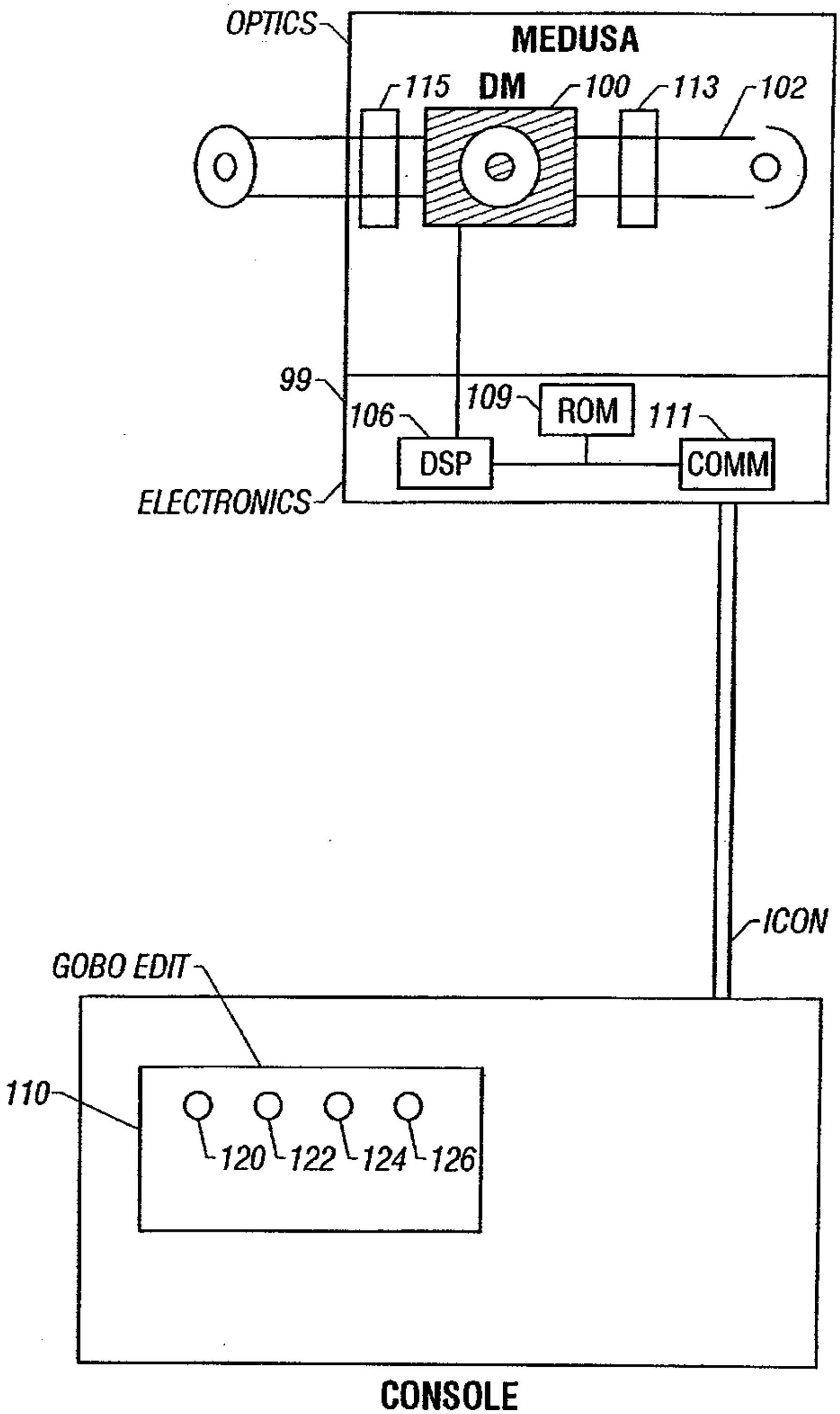
Primary Examiner—Loha Ben
(74) *Attorney, Agent, or Firm*—Fish & Richardson P.C.

(57) **ABSTRACT**

A special record format used for commanding light pattern shapes and addressable light pattern shape generator. The command format includes a first part which commands a specified gobo and second parts which command the characteristics of that gobo. The gobo is formed by making a default gobo based on the type and modifying that default gobo to fit the characteristics.

34 Claims, 6 Drawing Sheets





OPTICS
MEDUSA
115
DM
100
113
102
99
106
109
ROM
111
DSP
COMM
ELECTRONICS
ICON
GOBO EDIT
110
120
122
124
126
CONSOLE

FIG. 1

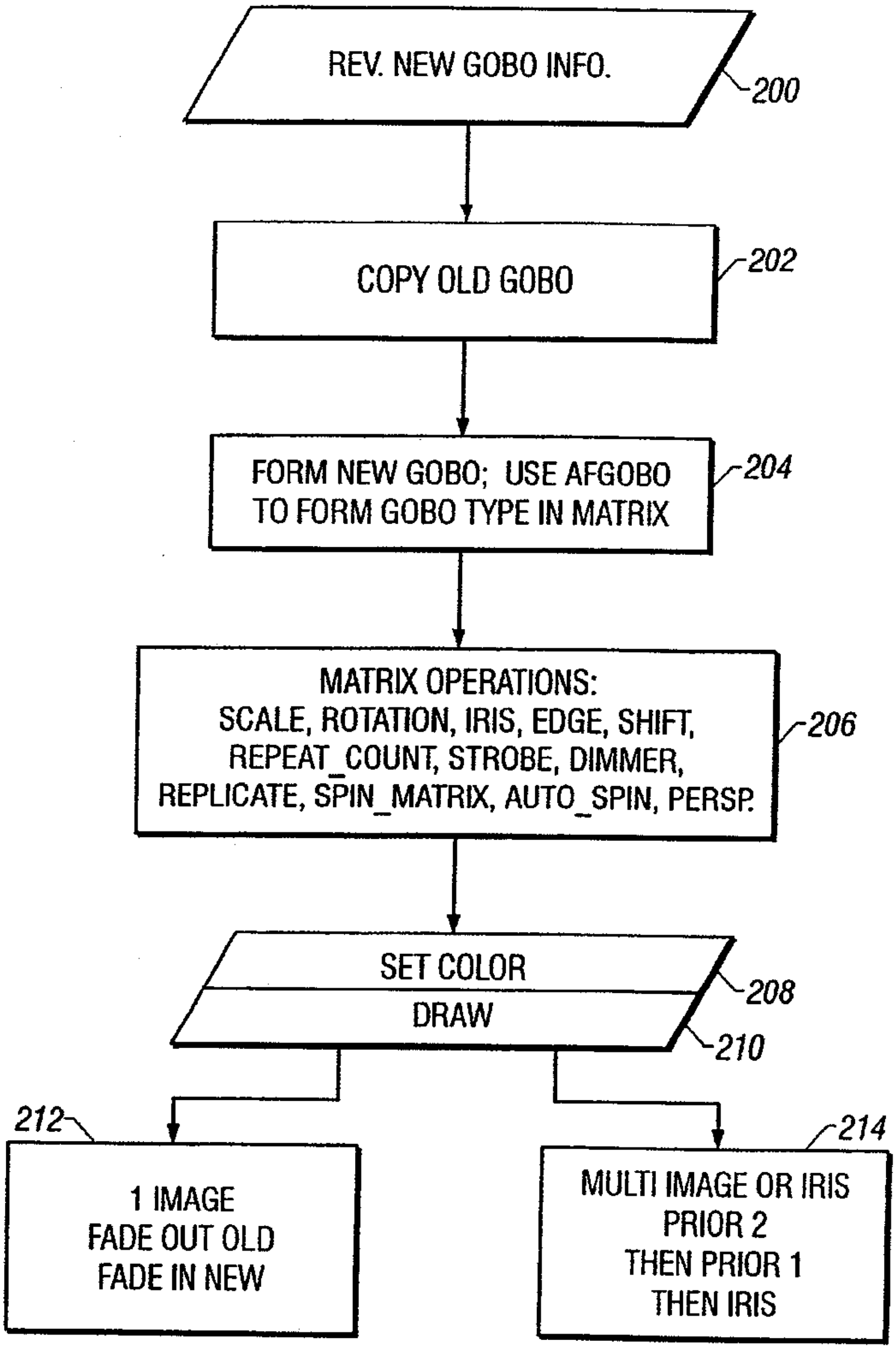

REV. NEW GOBO INFO.
200
COPY OLD GOBO
202
FORM NEW GOBO; USE AFGOBO TO FORM GOBO TYPE IN MATRIX
204
MATRIX OPERATIONS: SCALE, ROTATION, IRIS, EDGE, SHIFT, REPEAT_COUNT, STROBE, DIMMER, REPLICATE, SPIN_MATRIX, AUTO_SPIN, PERSP.
206
SET COLOR
208
DRAW
210
212
1 IMAGE FADE OUT OLD FADE IN NEW
214
MULTI IMAGE OR IRIS PRIOR 2 THEN PRIOR 1 THEN IRIS

FIG. 2

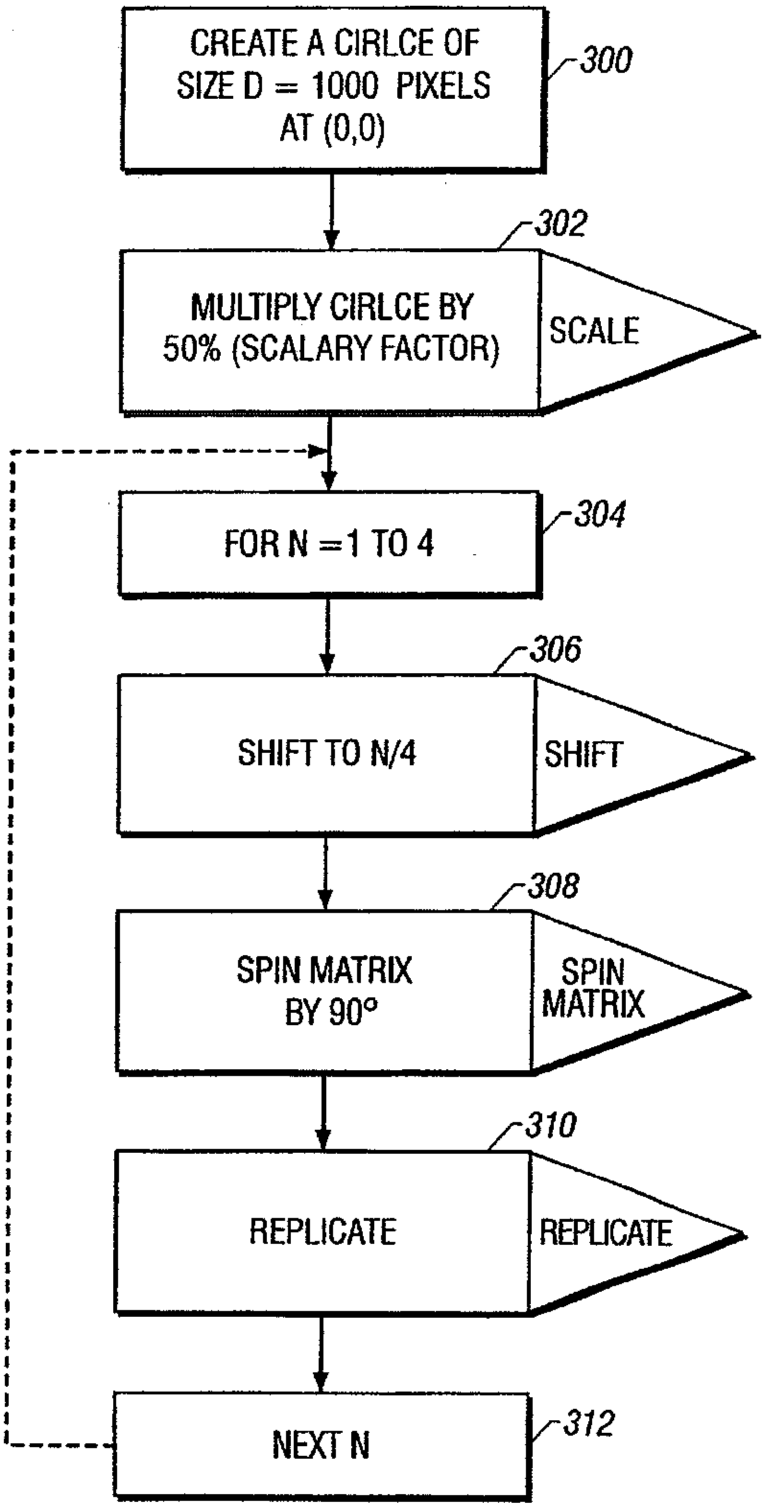

CREATE A CIRLCE OF SIZE D = 1000 PIXELS AT (0,0)
300
302
MULTIPLY CIRLCE BY 50% (SCALARY FACTOR)
SCALE
FOR N =1 TO 4
304
306
SHIFT TO N/4
SHIFT
308
SPIN MATRIX BY 90°
SPIN MATRIX
310
REPLICATE
REPLICATE
NEXT N
312

FIG. 3

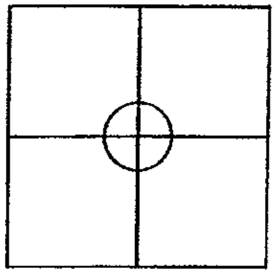


FIG. 4A

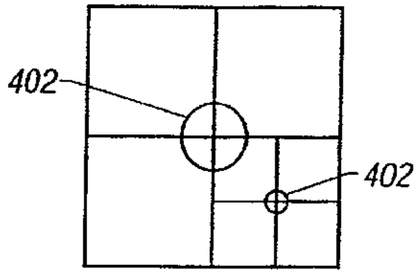


FIG. 4E

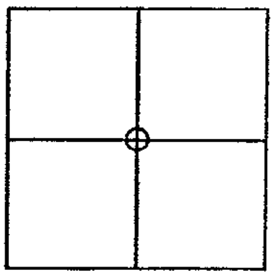


FIG. 4B

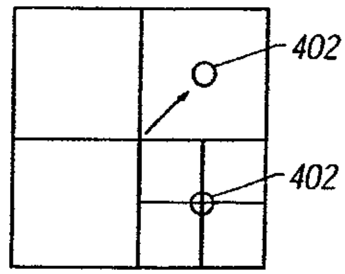


FIG. 4F

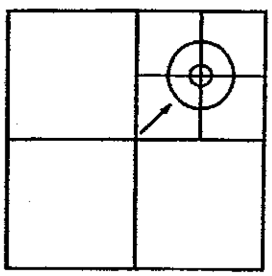


FIG. 4C

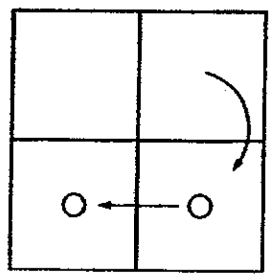


FIG. 4G

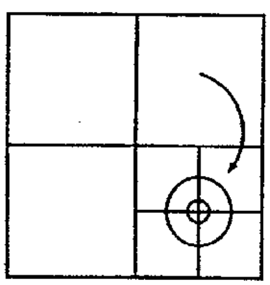


FIG. 4D

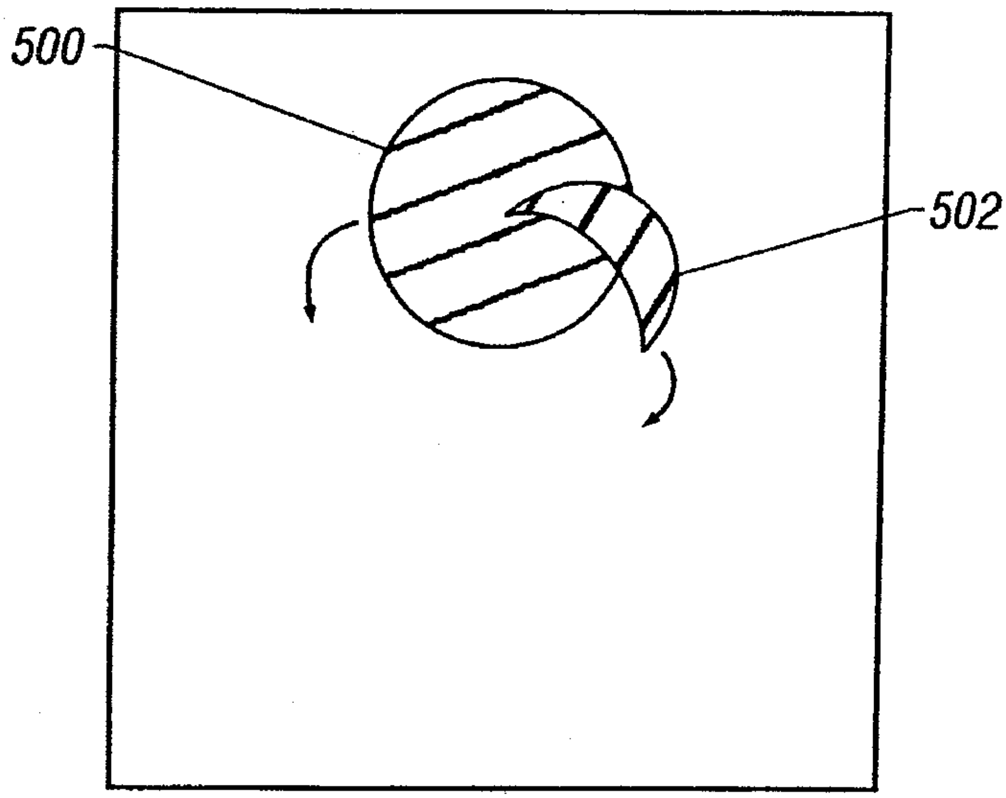
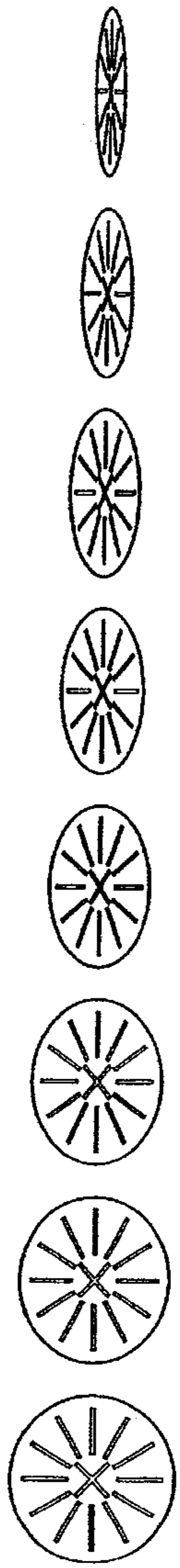


FIG. 5



8

7

6

5

4

3

2

1

FIG. 6

PIXEL BASED GOBO RECORD CONTROL FORMAT

This application is a continuation of U.S. application Ser. No. 09/500,393, filed Feb. 8, 2000, now U.S. Pat. No. 6,256,136, which is a continuation of U.S. application Ser. No. 09/145,314, filed Aug. 31, 1998, now U.S. Pat. No. 6,057,958, which claims priority from U.S. Provisional application Nos. 60/059,161, filed Sep. 17, 1997, and 60/065,133, filed Nov. 12, 1997.

FIELD

The present invention relates to a system of controlling light beam pattern ("gobo") shape in a pixilated gobo control system.

BACKGROUND

Commonly assigned patent application Ser. No. 08/854,353, now U.S. Pat. No. 6,188,933, the disclosure of which is herewith incorporated by reference, describes a stage lighting system which operates based on computer-provided commands to form special effects. One of those effects is control of the shape of a light pattern that is transmitted by the device. This control is carried out on a pixel-by-pixel basis, hence referred to in this specification as pixilated. Control is also carried out using an x-y controllable device. The preferred embodiment describes using a digital mirror device, but other x-y controllable devices such as a grating light valve, are also contemplated.

The computer controlled system includes a digital signal processor 106 which is used to create an image command. That image command controls the pixels of the x-y controllable device to shape the light that it is output from the device.

The system described in the above-referenced application allows unparalleled flexibility in selection of gobo shapes and movement. This opens an entirely new science of controlling gobos. The present inventors found that, unexpectedly, even more flexibility is obtained by a special control language for controlling those movements.

SUMMARY

The present disclosure defines a way of communicating with an x-y controllable device to form special electronic light pattern shapes. More specifically, the present application describes using a control language to communicate with an electronic gobo in order to reposition part or all of the image that is shaping the light.

BRIEF DESCRIPTION OF THE DRAWINGS

These and other aspects of the invention will now be described with reference to the attached drawings, in which:

FIG. 1 shows a block diagram of the basic system operating the embodiment;

FIG. 2 shows a basic flowchart of operation;

FIG. 3 shows a flowchart of forming a replicating circles type gobo;

FIG. 4A through 4G show respective interim results of carrying out the replicating circles operation;

FIG. 5 shows the result of two overlapping gobos rotating in opposite directions; and

FIGS. 6(1) through 6(8) show a z-axis flipping gobo.

DESCRIPTION OF THE PREFERRED EMBODIMENT

FIG. 1 shows a block diagram of the hardware used according to the preferred embodiment. As described above, this system uses a digital mirror device 100, which has also been called a digital mirror device ("DMD") and a digital light processor device ("DLP"). More generally, any system which allows controlling shape of light on a pixel basis, including a grating light valve, could be used as the light shaper. This light shaper forms the shape of light which is transmitted. FIG. 1 shows the light being transmitted as 102, and shows the transmitted light. The information for the digital mirror 100 is calculated by a digital signal processor 106. Information is calculated based on local information stored in the lamp, e.g., in ROM 109, and also in information which is received from the console 104 over the communication link.

The operation is commanded according to a format.

The preferred data format provides 4 bytes for each of color and gobo control information.

The most significant byte of gobo control data, ("dfGobo") indicates the gobo type. Many different gobo types are possible. Once a type is defined, the gobo formed from that type is represented by a number. That type can be edited using a special gobo editor described herein. The gobo editor allows the information to be modified in new ways, and forms new kinds of images and effects.

The images which are used to form the gobos may have variable and/or moving parts. The operator can control certain aspects of these parts from the console via the gobo control information. The type of gobo controls the gobo editor to allow certain parameters to be edited.

The examples given below are only exemplary of the types of gobo shapes that can be controlled, and the controls that are possible when using those gobo shapes. Of course, other controls of other shapes are possible and predictable based on this disclosure.

A first embodiment is the control of an annulus, or "ring" gobo. The DMD 100 in FIG. 1 is shown with the ring gobo being formed on the DMD. The ring gobo is type 000A. When the gobo type 0A is enabled, the gobo editor 110 on the console 104 is enabled and the existing gobo encoders 120, 122, 124, and 126 are used. The gobo editor 110 provides the operator with specialized control over the internal and the external diameters of the annulus, using separate controls in the gobo editor.

The gobo editor and control system also provides other capabilities, including the capability of timed moves between different edited parameters. For example, the ring forming the gobo could be controlled to be thicker. The operation could then effect a timed move between these "preset" ring thicknesses. Control like this cannot even be attempted with conventional fixtures.

Another embodiment is a composite gobo with moving parts. These parts can move though any path that are programmed in the gobo data itself. This is done in response to the variant fields in the gobo control record, again with timing. Multiple parts can be linked to a single control allowing almost unlimited effects.

Another embodiment of this system adapts the effect for an "eye" gobo, where the pupil of the eye changes its position (look left, look right) in response to the control.

Yet another example is a Polygon record which can be used for forming a triangle or some other polygonal shape.

The control can be likened to the slider control under a QuickTime movie window, which allows you to manually move to any point in the movie. However, our controls need not be restricted to timelines.

Even though such moving parts are used, scaling and rotation on the gobo is also possible.

The following type assignments are contemplated:

00-0F=FixedGobo (with no "moving parts")

10-1F=SingleCntrl (with 1 "moving part")

20-2F=DoubleCntrl (with 2 "moving parts")

30-FF=undefined, reserved.

The remaining control record bytes for each type are defined as follows:

Byte:	dfGobo2	dfGobo3	dfGobo4	#gobos/type,	total memory
FixedGobo:	ID[23:16]	ID[15:8]	ID[7:0]	16M/type	256M
SingleCntrl:	ID[15:8]	ID[7:0]	control#1	64k/type	1M
DoubleCntrl:	ID[7:0]	control#2	control#1	256/type	4k

As can be seen from this example, this use of the control record to carry control values does restrict the number of gobos which can be defined of that type, especially for the 2-control type.

Console Support

The use of variant part gobos requires no modifications to existing console software for the ICON (7M) console. The Gobo editor in current ICON software already provides 4 separate encoders for each gobo. These translate directly to the values of the 4 bytes sent in the communications data packet as follows:

Byte:	dfGobo	dfGobo2	dfGobo3	dfGobo4
Enc:	TopRight	MidRight	BotRight	BotLeft
FixedGobo:		ID[23:16]	ID[15:8]	ID[7:0]
SingleCntrl:		ID[15:8]	ID[7 0]	control#1
DoubleCntrl:		ID[7:0]	control#2	control#1

These values would be part of a preset gobo, which could be copied as the starting point.

Once these values are set, the third and fourth channels automatically become the inner/outer radius controls. Using two radii allows the annulus to be turned "inside out".

Each control channel's data always has the same meaning within the console. The console treats these values as simply numbers that are passed on. The meanings of those numbers, as interpreted by the lamps change according to the value in dfGobo.

The lamp will always receives all 4 bytes of the gobo data in the same packet. Therefore, a "DoubleCntrl" gobo will always have the correct control values packed along with it.

Hence, the console needs no real modification. If a "soft" console is used, then name reassignments and/or key reassignments may be desirable.

Timing

For each data packet, there is an associated "Time" for gobo response. This is conventionally taken as the time allotted to place the new gobo in the light gate. This delay has been caused by motor timing. In this system, variant gobo, the control is more dynamically used. If the non-variant parts of the gobo remain the same, then it is still the same gobo, only with control changes. Then, the time value is interpreted as the time allowed for the control change.

Since different gobo presets (in the console) can reference the same gobo, but with different control settings, this allows easily programmed timed moves between different annuli, etc.

Internal Workings

When the gobo command data is extracted from the packet at the lamp, the dfGobo byte is inspected first, to see if either dfGobo3 or dfGobo4 are significant in selecting the image. In the case of the "Cntrl" variants, one or both of these bytes is masked out, and the resulting 32-bit number is used to search for a matching gobo image (by Gobo_ID) in the library stored in the lamp's ROM **109**.

If a matching image is found, and the image is not already in use, then the following steps are taken:

1) The image data is copied into RAM, so that its fields may be modified by the control values. This step will be skipped if the image is currently active.

2) The initial control values are then recovered from the data packet, and used to modify certain fields of the image data, according to the control records.

3) The image is drawn on the display device, using the newly-modified fields in the image data.

If the image is already in use, then the RAM copy is not altered. Instead, a time-sliced task is set up to slew from the existing control values to those in the new data packet, in a time determined by the new data packet.

At each vertical retrace of the display, new control values are computed, and steps 2 (using the new control values) and 3 above are repeated, so that the image appears modified with time.

The Image Data Records

All images stored in the lamp are in a variant record format:

Header:

Length 32 bits, offset to next gobo in list.

Gobo _ID 32 bits, serial number of gobo.

Gobo records:

Length 32 bits, offset to next record.

Opcode 16 bits, type of object to be drawn.

Data Variant part—data describing object.

_Length 32 bits, offset to next record.

Opcode 16 bits, type of object to be drawn.

Data Variant part—data describing object.

_EndMarker 64 bits, all zeroes—indicates end of gobo data.

+Next gobo, or End Marker, indicating end of gobo list.

Gobos with controls are exactly the same, except that they contain control records, which describe how the control values are to affect the gobo data. Each control record contains the usual length and Opcode fields, and a field containing the control number (1 or 2).

These are followed by a list of "field modification-"records. Each record contains information about the offset (from the start of the gobo data) of the field, the size (8, 16 or 32 bits) of the field, and how its value depends on the control value.

Length	32 bits, offset to next record
Opcode	16 bits = control_record (constant)
CntrlNum	16 bits = 1 or 2 (control number)
	/* field modification record #1 */
Address	16 bits, offset from start of gobo to affected field.
Flags	16 bits, information about field (size, signed,

-continued

	etc)
Scale	16 bits, scale factor applied to control before use
ZPoint	16 bits, added to control value after scaling.
	/* field modification record #2 */
Address	16 bits, offset from start of gobo to affected field.
Flags	16 bits, information about field (size, signed, etc)
Scale	16 bits, scale factor applied to control before use
ZPoint	16 bits, added to control value after scaling.

As can be seen, a single control can have almost unlimited effects on the gobo, since ANY values in the data can be modified in any way, and the number of field modification records is almost unlimited.

Note that since the control records are part of the gobo data itself, they can have intimate knowledge of the gobo structure. This makes the hard-coding of field offsets acceptable.

In cases where the power offered by this simple structure is not sufficient, a control record could be defined which contains code to be executed by the processor. This code would be passed parameters, such as the address of the gobo data, and the value of the control being adjusted.

Example records.

The Annulus record has the following format:

Length	32 bits
Opcode	16 bits, = type_annulus
Pad	16 bits, unused
Centre_x	16 bits, x coordinate of centre
Centre_y	16 bits, y coordinate of centre
OuterRad	16 bits, outside radius (the radii get swapped when drawn if their values are in the wrong order)
InnerRad	16 bits, inside radius

It can be seen from this that it is easy to "target" one of the radius parameters from a control record. Use of two control records, each with one of the radii as a target, would provide full control over the annulus shape.

Note that if the center point coordinates are modified, the annulus will move around the display area, independent of any other drawing elements in the same gobo's data.

The Polygon record for a triangle has this format:

The Polygon record for a triangle has this format:

Length	32 bits
Opcode	16 bits, = type_polygon
Pad	16 bits, vertex count = 3
Centre_x	16 bits, x coordinate of vertex
Centre_y	16 bits, y coordinate of vertex
Centre_x	16 bits, x coordinate of vertex
Centre_y	16 bits, y coordinate of vertex
Centre_x	16 bits, x coordinate of vertex
Centre_y	16 bits, y coordinate of vertex

It is easy to modify any of the vertex coordinates, producing distortion of the triangle.

The gobo data can contain commands to modify the drawing environment, by rotation, scaling, offset, and color control, the power of the control records is limitless.

Second Embodiment

This second embodiment provides further detail about implementation once the gobo information is received.

Gobo information is, at times, being continuously calculated by DSP **106**. The flowchart of FIG. **2** shows the handling operation that is carried out when new gobo information is received.

At step **200**, the system receives new gobo information. In the preferred embodiment, this is done by using a communications device **111** in the lamp **99**. The communications device is a mailbox which indicates when new mail is received. Hence, the new gobo information is received at step **200** by determining that new mail has been received.

At step **202**, the system copies the old gobo and switches pointers. The operation continues using the old gobo until the draw routine is called later on.

At step **204**, the new information is used to form a new gobo. The system uses a defined gobo ("dfGobo") as discussed previously which has a defined matrix. The type dfGobo is used to read the contents from the memory **109** and thereby form a default image. That default image is formed in a matrix. For example, in the case of an annulus, a default size annulus can be formed at position **0,0** in the matrix. An example of forming filled balls is provided herein.

Step **206** represents calls to subroutines. The default gobo is in the matrix, but the power of this system is its ability to very easily change the characteristics of that default gobo. In this embodiment, the characteristics are changed by changing the characteristics of the matrix and hence, shifting that default gobo in different ways. The matrix operations, which are described in further detail herein, include scaling the gobo, rotation, iris, edge, strobe, and dimmer. Other matrix operations are possible. Each of these matrix operations takes the default gobo, and does something to it.

For example, scale changes the size of the default gobo rotation rotates the default gobo by a certain amount. Iris simulates an iris operation by choosing an area of interest, typically circular, and erasing everything outside that area of interest. This is very easily done in the matrix, since it simply defines a portion in the matrix where all black is written.

Edge effects carry out certain effects on the edge such as softening the edge. This determines a predetermined thickness, which is translated to a predetermined number of pixels, and carries out a predetermined operation on the number of pixels. For example, for a 50% edge softening, every other pixel can be turned off. The strobe is in effect that allows all pixels to be turned on and off at a predetermined frequency, i.e., 3 to 10 times a second. The dimmer allows the image to be made dimmer by turning off some of the pixels at predetermined times.

The replicate command forms another default gobo, to allow two different gobos to be handled by the same record. This will be shown with reference to the exemplary third embodiment showing balls. Each of those gobos is then handled as the same unit and the entirety of the gobos can be, for example, rotated. The result of step **206** and all of these subroutines that are called is that the matrix includes information about the bits to be mapped to the digital mirror **100**.

At step **208**, the system then obtains the color of the gobos from the control record discussed previously. This gobo color is used to set the appropriate color changing circuitry **113** and **115** in the lamp **99**. Note that the color changing circuitry is shown both before and after the digital mirror **100**. It should be understood that either of those color changing circuits could be used by itself.

At step **210**, the system calls the draw routine in which the matrix is mapped to the digital mirror. This is done in

different ways depending on the number of images being used. Step **212** shows the draw routine for a single image being used as the gobo. In that case, the old gobo, now copied as shown in step **202**, is faded out while the new gobo newly calculated is faded in. Pointers are again changed so that the system points to the new gobo. Hence, this has the effect of automatically fading out the old gobo and fading in the new gobo.

Step **214** schematically shows the draw routine for a system with multiple images for an iris. In that system, one of the gobos is given priority over the other. If one is brighter than the other, then that one is automatically given priority. The one with priority **2**, the lower priority one, is written first. Then the higher priority gobo is written. Finally, the iris is written which is essentially drawing black around the edges of the screen defined by the iris. Note that unlike a conventional iris, this iris can take on many different shapes. The iris can take on not just a circular shape, but also an elliptical shape, a rectangular shape, or a polygonal shape. In addition, the iris can rotate when it is non-circular so that for the example of a square iris, the edges of the square can actually rotate.

Returning to step **206**, in the case of a replicate, there are multiple gobos in the matrix. This allows the option of spinning the entire matrix, shown as spin matrix.

An example will now be described with reference to the case of repeating circles. At step **200**, the new gobo information is received indicating a circle. This is followed by the other steps of **202** where the old gobo is copied, and **204** where the new gobo is formed. The specific operation forms a new gobo at step **300** by creating a circle of size diameter equals 1000 pixels at origin **00**. This default circle is automatically created. FIG. **4**A shows the default gobo which is created, a default size circle at **00**. It is assumed for purposes of this operation that all of the circles will be the same size.

At step **302**, the circle is scaled by multiplying the entire circle by an appropriate scaling factor. Here, for simplicity, we are assuming a scaling factor of 50% to create a smaller circle. The result is shown in FIG. **4**B. A gobo half the size of the gobo of FIG. **4**A is still at the origin. This is actually the scale of the subroutine as shown in the right portion of step **302**. Next, since there will be four repeated gobos in this example, a four-loop is formed to form each of the gobos at step **304**. Each of the gobos is shifted in position by calling the matrix operator shift. In this example, the gobo is shifted to a quadrant to the upper right of the origin. This position is referred to as over **4** in the FIG. **3** flowchart and results in the gobo being shifted to the center portion of the top right quadrant as shown in FIG. 4C. This is again easily accomplished within the matrix by moving the appropriate values. At step **308**, the matrix is spun by 90 degrees in order to put the gobo in the next quadrant as shown in FIG. 4D in preparation for the new gobo being formed into the same quadrant. Now the system is ready for the next gobo, thereby calling the replicate command which quite easily creates another default gobo circle and scales it. The four-loop is then continued at step **312**.

The replicate process is shown in FIG. 4E where a new gobo **402** is formed in addition to the existing gobo **400**. The system then passes again through the four-loop, with the results being shown in the following figures. In FIG. 4F, the new gobo **402** is again moved to the upper right quadrant (step **306**). In FIG. 4G, the matrix is again rotated to leave room for a new gobo in the upper right quadrant. This continues until the end of the four-loop. Hence, this allows each of the gobos to be formed.

Since all of this is done in matrix operation, it is easily programmable into the digital signal processor. While the above has given the example of a circle, it should be understood that this scaling and moving operation can be carried out for anything. The polygons, circles, annulus, and everything else is easily scaled.

The same operation can be carried out with the multiple parameter gobos. For example, for the case of a ring, the variable takes the form annulus (inner R, outer R, x and y). This defines the annulus and turns of the inner radius, the outer radius, and x and y offsets from the origin. Again, as shown in step **3**, the annulus is first written into the matrix as a default size, and then appropriately scaled and shifted. In terms of the previously described control, the ring gobo has two controls: control **1** and control **2** defined the inner and outer radius.

Each of these operations is also automatically carried out by the command repeat count which allows easily forming the multiple position gobo of FIGS. **4**A–**4**G. The variable auto spin defines a continuous spin operation. The spin operation commands the digital signal processor to continuously spin the entire matrix by a certain amount each time.

One particularly interesting feature available from the digital mirror device is the ability to use multiple gobos which can operate totally separately from one another raises the ability to have different gobos spinning in different directions. When the gobos overlap, the processor can also calculate relative brightness of the two gobos. In addition, one gobo can be brighter than the other. This raises the possibility of a system such as shown in FIG. **5**. Two gobos are shown spinning in opposite directions: the circle gobo **500** is spinning the counterclockwise direction, while the half moon gobo **502** is spinning in the clockwise direction. At the overlap, the half moon gobo which is brighter than the circle gobo, is visible over the circle gobo. Such effects were simply not possible with previous systems. Any matrix operation is possible, and only a few of those matrix operations have been described herein.

A final matrix operation to be described is the perspective transformation. This defines rotation of the gobo in the Z axis and hence allows adding depth and perspective to the gobo. For each gobo for which rotation is desired, a calculation is preferably made in advance as to what the gobo will look like during the Z axis transformation. For example, when the gobo is flipping in the Z axis, the top goes back and looks smaller while the front comes forward and looks larger. FIGS. **6(1)**–**(8)** show the varying stages of the gobo flipping. In FIG. **6(8)**, the gobo has its edge toward the user. This is shown in FIG. **6(8)** as a very thin line, e.g., three pixels wide, although the gobo could be zero thickness at this point. Automatic algorithms are available for such Z axis transformation, or alternatively a specific Z axis transformation can be drawn and digitized automatically to enable a custom look.

Although only a few embodiments have been described in detail above, other embodiments are contemplated by the inventor and are intended to be encompassed within the following claims. In addition, other modifications are contemplated and are also intended to be covered.

What is claimed is:

1. A lighting console, comprising:

a plurality of control parts including a first control part that controls a gobo type and a second control part that controls characteristics of a gobo selected by the first control part,

a processor, responsive to said first control part to enable different characteristics for said second control part based on the type of said first control part; and

an output signal protection part, which produces a record indicating both said gobo type and said gobo characteristics.

2. A console as in claim 1, wherein said console provides an output of an alphanumeric code indicative of said gobo type and parameters for the specified gobo type.

3. A console as in claim 2, wherein said console includes a plurality of encoders, and said encoders are selectively enabled based on a selected gobo type.

4. A console as in claim 1, wherein a round gobo type can be selected, and one of said controllable characteristics is a diameter of said round gobo type.

5. A console as in claim 1, wherein an annular gobo type can be selected, and one of said controllable characteristics is a thickness of the annulus.

6. A console as in claim 1, further comprising enabling preset but different characteristics for a specified gobo.

7. A console as in claim 6, wherein one characteristic of said second control part is a timing of changing said gobo between said different characteristics.

8. A console as in claim 1, wherein said second control part enables controlling a time to change between different characteristics.

9. A method, comprising:

using a lighting console to define a gobo type having editable parameters;

selecting two different parameter values; and

defining a time for said gobo type to change its appearance between said two different parameter values.

10. A method as in claim 9, wherein said editable parameters include a thickness of a gobo part.

11. A method as in claim 9, wherein said editable parameters include a diameter of a gobo part.

12. A method as in claim 9, wherein said gobo type is a composite gobo with moving parts, and said parameter values define paths for the respective moving parts.

13. A method, comprising:

using a lighting console to define first and second lighting objects collectively forming a composite gobo; and

defining movement information as part of gobo data; and

forming the gobo using said objects and said movement information.

14. A method as in claim 13, wherein said forming comprises producing a signal and applying said signal to an electrically controllable light pattern generator to form the gobo.

15. A method as in claim 13, further comprising using the lighting console to define timing information for the first and second objects.

16. A method as in claim 13, wherein said forming comprises forming a single record which defines said objects and said movement information.

17. A method, comprising:

selecting a gobo from a library of different electronic gobos; and

characterizing said gobo according to a number of moving parts within the gobo.

18. A method as in claim 17, further comprising outputting data according to a type assignment which includes a code indicative of said number of moving parts.

19. A method as in claim 18, wherein said type assignment includes a first type assignment with no moving parts, a second type assignment with one moving part, and a third type assignment with more than one moving part.

20. A method as in claim 17, further comprising, for a gobo with at least one moving part, selecting a time for response for said gobo moving part.

21. A method, comprising:

defining a gobo type in a lighting console, and sending an identification signal indicative of said gobo type from said lighting console to an electronically controlled light which is remote from said lighting console; and

using said identification signal to search a library within said electronically controlled light for a matching gobo image and to use said matching gobo image to shape an output light from said electronically controlled light.

22. A method as in claim 21, wherein said using comprises receiving a gobo control signal in said electronically controlled light, and using only some of a plurality of bits of said electronically controlled signal to search for a matching gobo image.

23. A method as in claim 21, wherein said using comprises copying a matching gobo image from said library to a random access memory within said electronically controlled light.

24. A method as in claim 23, further comprising forming modification information for said gobo type in said lighting console, and using said modification information to modify said gobo image in said random access memory.

25. A method as in claim 24, further comprising using said gobo image in said random access memory to draw an image on the display device.

26. A method as in claim 24, wherein said display device is an electronically alterable, pixel-selectable digital mirror device.

27. A method as in claim 24, further comprising obtaining new modification information, and using said new modification information to modify said gobo image in said random access memory.

28. A method as in claim 27, further comprising using said gobo image in said random access memory to draw an image on the display device which is periodically retraced, and calculating new values at each time of retrace.

29. A method as in claim 24, further comprising obtaining new modification information, forming a new image using said new modification information, and slewing from said existing gobo image to a new gobo image represented by said new modification information.

30. A method, comprising:

defining gobo information using an electronic gobo controller element to select a specified gobo type and parameters associated with said gobo type;

forming a gobo shape of a default size of said specified gobo type; and

modifying said specified gobo type according to set parameters.

31. A method as in claim 30, wherein said default size gobo shape is formed at the specified position.

32. A method as in claim 30, further comprising enabling said default size gobo to be replicated.

33. A method, comprising:

defining multiple shapes which collectively define the gobo to be used in displaying an image;

setting priorities for the multiple shapes; and

displaying the multiple shapes according to the priorities.

34. A method as in claim 33, wherein said displaying comprises always displaying a highest priority shape at an intersection between said highest priority shape and a lower priority shape.

* * * * *

EXHIBIT M

US006515435B2

(12) **United States Patent**
Hughes et al.

(10) **Patent No.: US 6,515,435 B2**
(45) **Date of Patent: Feb. 4, 2003**

(54) **PIXEL MIRROR BASED STAGE LIGHTING SYSTEM**

(75) Inventors: **Michael Hughes**, Wolverhampton (GB); **William Hewlett**, Birmingham (GB); **Ian Clarke**, Walsall (GB)

(73) Assignee: **Light and Sound Design, Ltd.**, Birmingham (GB)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 14 days.

(21) Appl. No.: **09/756,037**

(22) Filed: **Jan. 5, 2001**

(65) **Prior Publication Data**

US 2002/0070686 A1 Jun. 13, 2002

Related U.S. Application Data

(62) Division of application No. 09/145,313, filed on Aug. 31, 1998, now Pat. No. 6,208,087.

(51) **Int. Cl.**[7] **H05B 37/00**

(52) **U.S. Cl.** **315/292**; 315/297; 340/641; 340/642

(58) **Field of Search** 340/640, 642, 340/641, 643; 315/105, 106, 107, 294, 297, 292

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,588,927 A	*	5/1986	Kanno et al.	315/307
5,868,666 A	*	2/1999	Okada et al.	600/118
6,268,799 B1	*	7/2001	Miyashita et al.	340/641

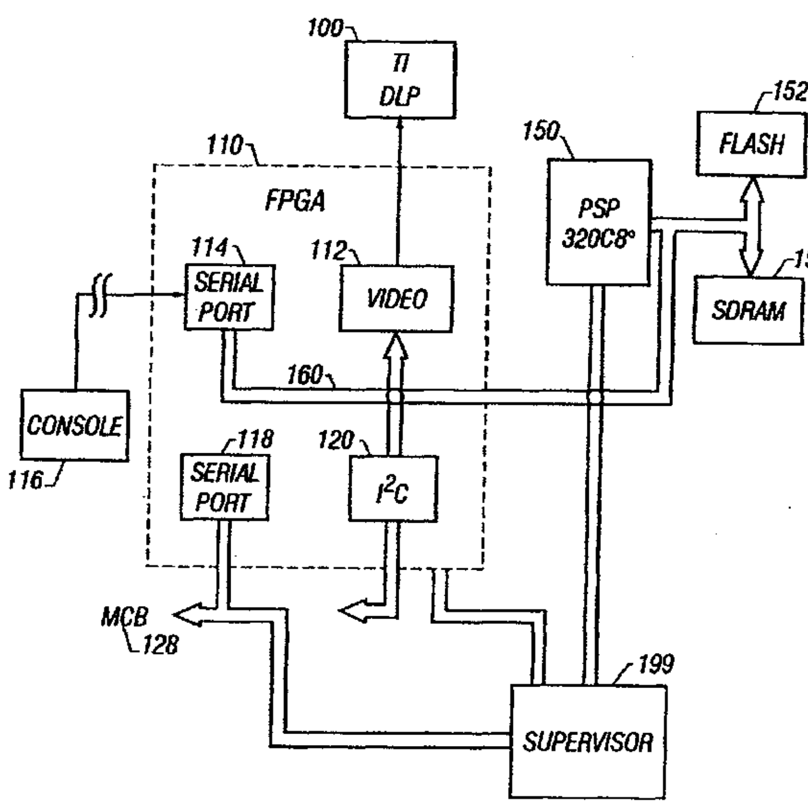
* cited by examiner

Primary Examiner—David Vu
(74) *Attorney, Agent, or Firm*—Fish & Richardson P.C.

(57) **ABSTRACT**

A lighting system with a digital micromirror device which forms a digital gobo system that is controllable on a pixel level. The device includes a number of sensitive electronic elements. The device is controlled by two different controllers: a digital signal processor which is effectively the number crunching portion, and a controller, which controls the functions of the light. The controller also logs errors into a registry, such as a non-volatile memory. A field programmable gate array is used for a number of purposes, including to form the ports. One of those ports is a tech port, which is used to detect status of the light such as faults.

8 Claims, 6 Drawing Sheets



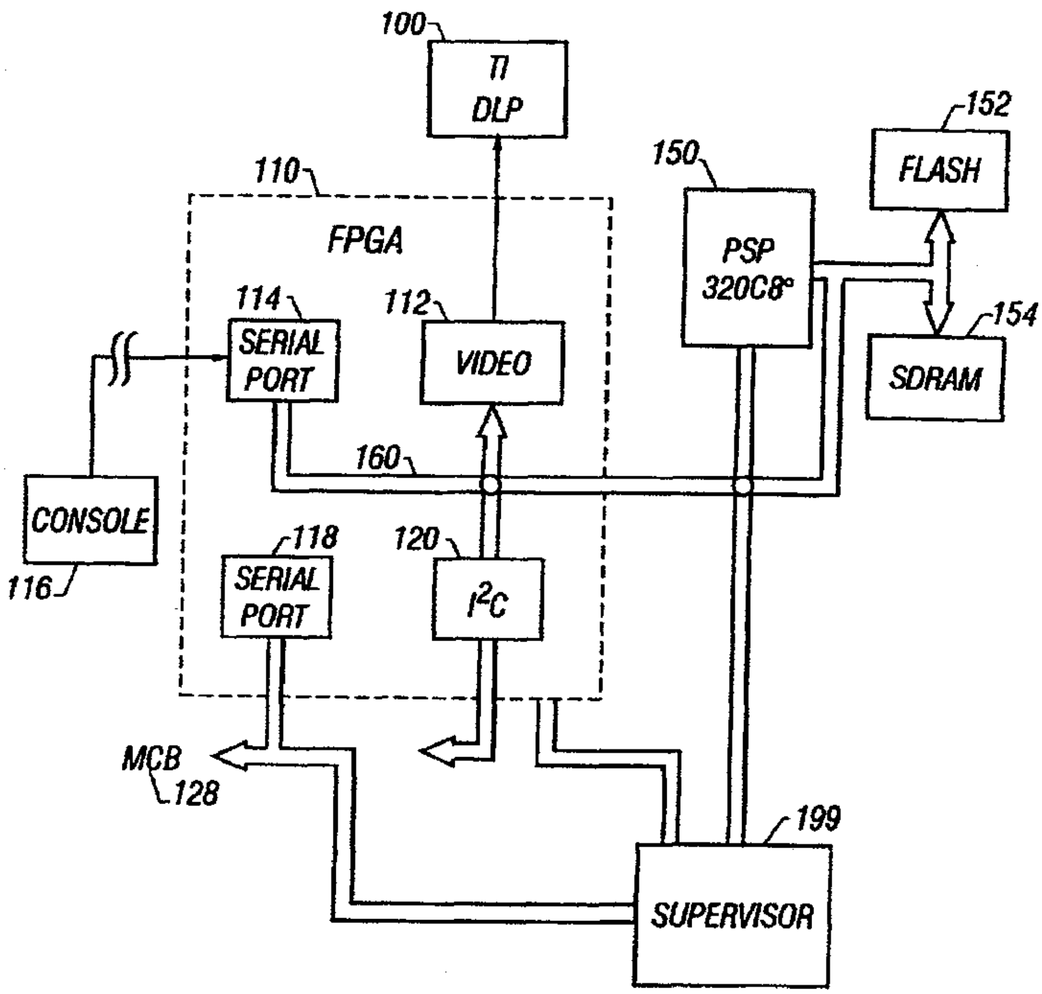
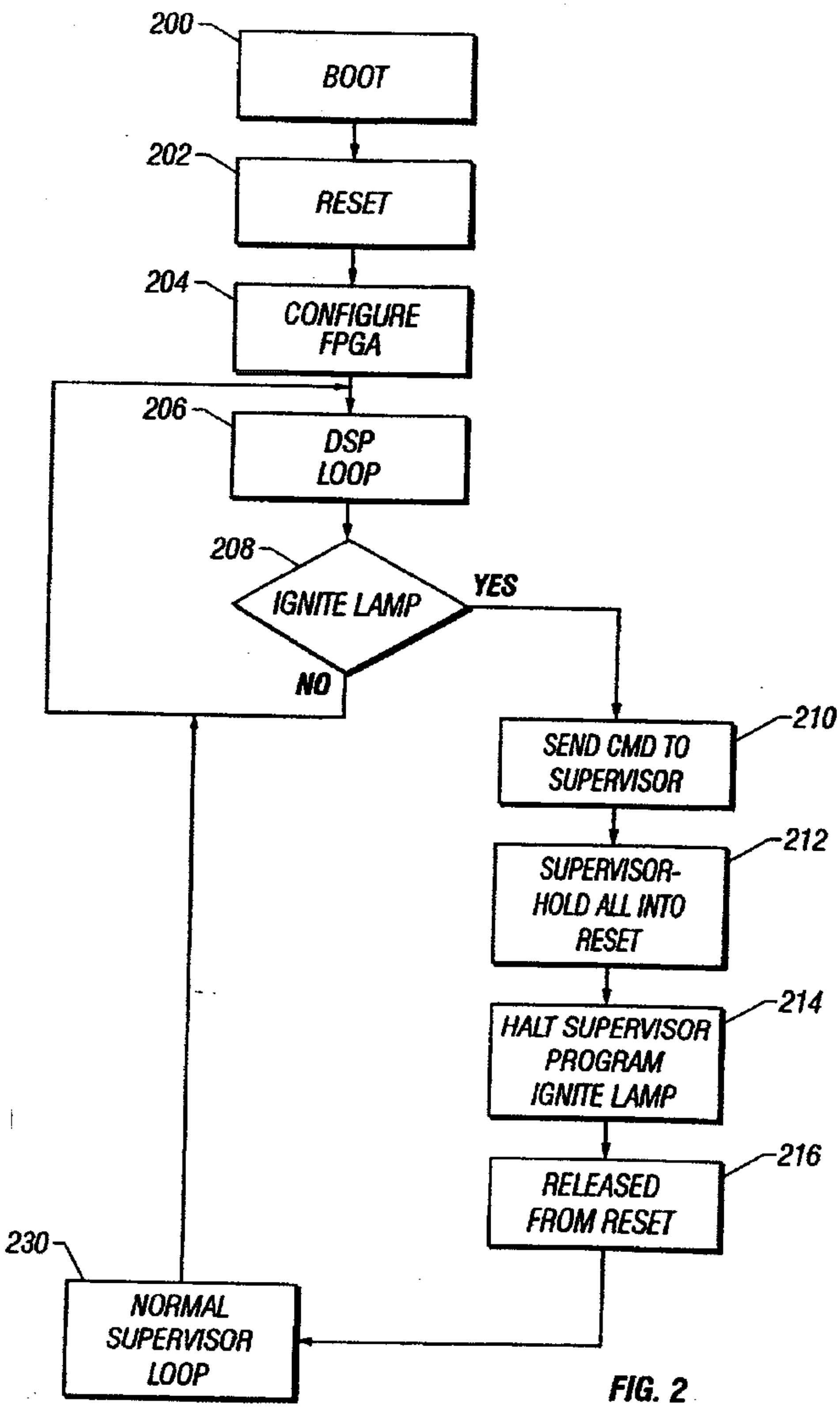

100
TI
DLP
110
FPGA
150
PSP
320C8°
152
FLASH
154
SDRAM
114
SERIAL
PORT
112
VIDEO
160
CONSOLE
116
118
SERIAL
PORT
120
I²C
MCB
128
199
SUPERVISOR

FIG. 1



200
BOOT
202
RESET
204
CONFIGURE FPGA
206
DSP LOOP
208
IGNITE LAMP
YES
NO
SEND CMD TO SUPERVISOR
210
SUPERVISOR- HOLD ALL INTO RESET
212
HALT SUPERVISOR PROGRAM IGNITE LAMP
214
RELEASED FROM RESET
216
230
NORMAL SUPERVISOR LOOP

FIG. 2

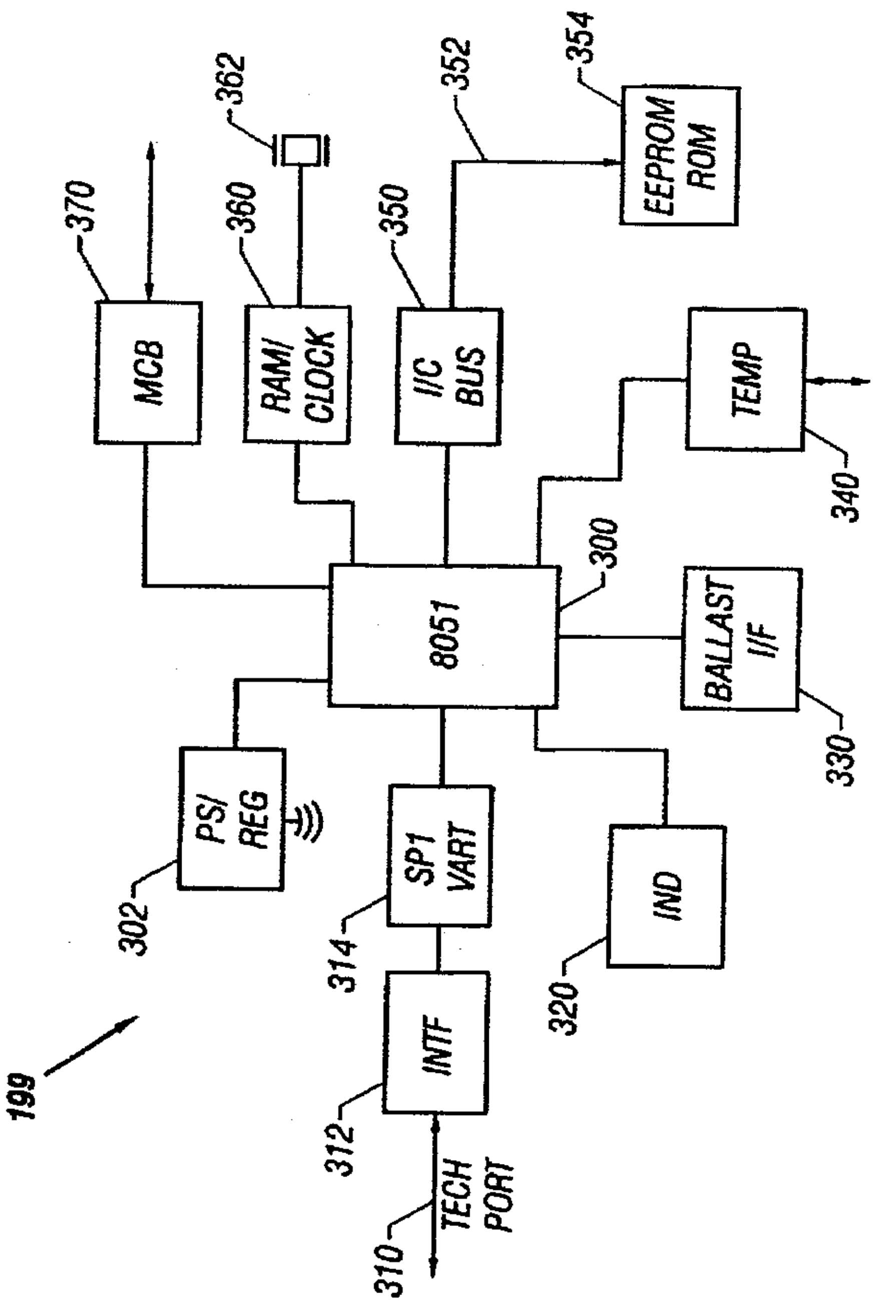

199
370
MCB
360
RAM/
CLOCK
362
350
I/C
BUS
352
354
EEPROM
ROM
TEMP
340
8051
300
BALLAST
I/F
330
PS/
REG
302
SP1
VART
314
IND
320
INTF
312
TECH
PORT
310

FIG.3

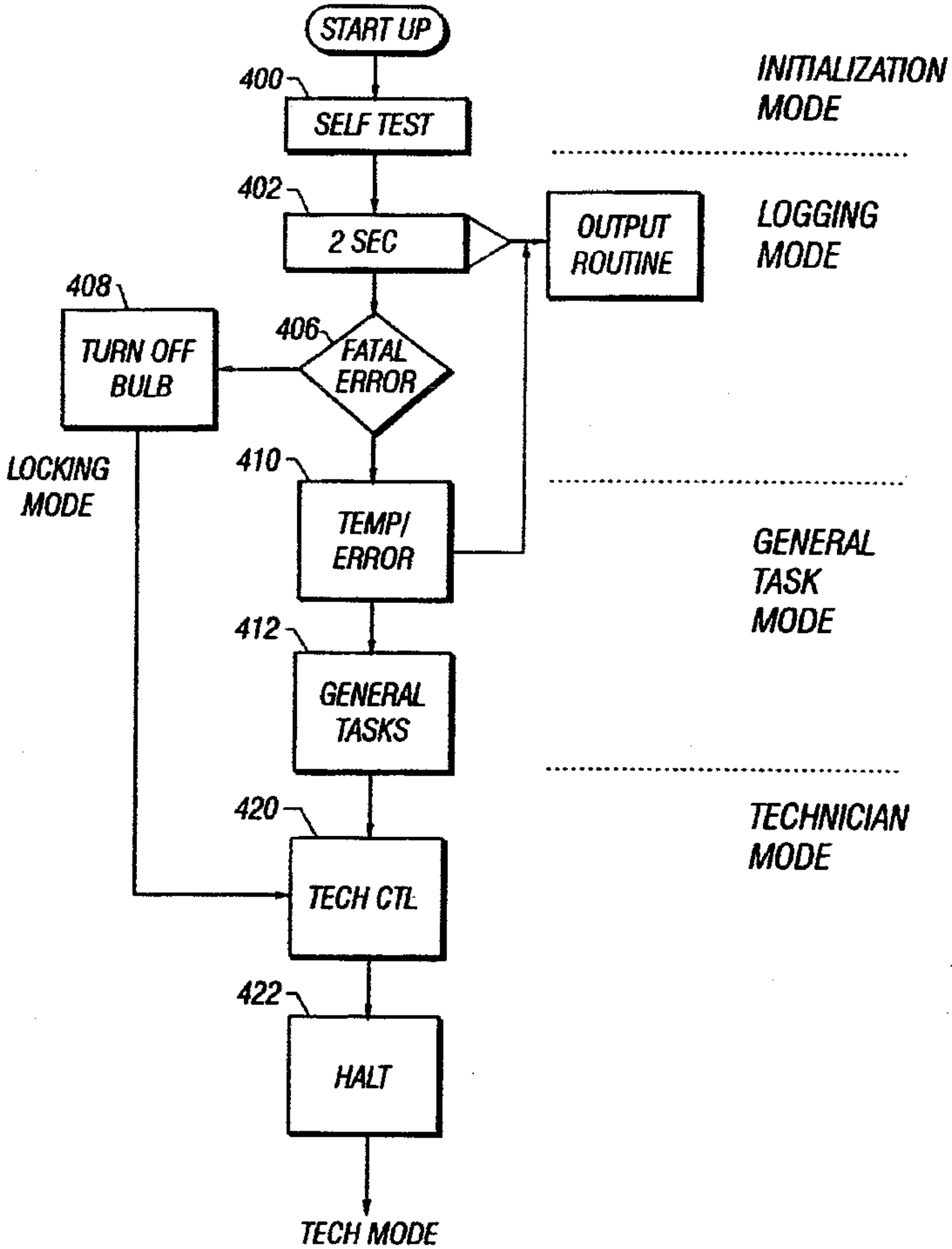

START UP
400
SELF TEST
INITIALIZATION MODE
402
2 SEC
OUTPUT ROUTINE
LOGGING MODE
408
TURN OFF BULB
406
FATAL ERROR
LOCKING MODE
410
TEMP/ ERROR
GENERAL TASK MODE
412
GENERAL TASKS
420
TECH CTL
TECHNICIAN MODE
422
HALT
TECH MODE

FIG.4

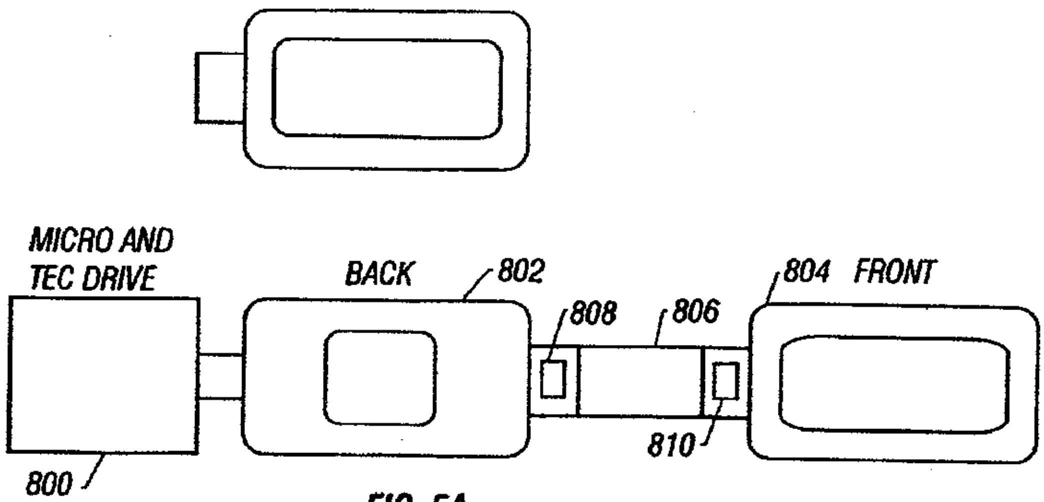


FIG. 5A

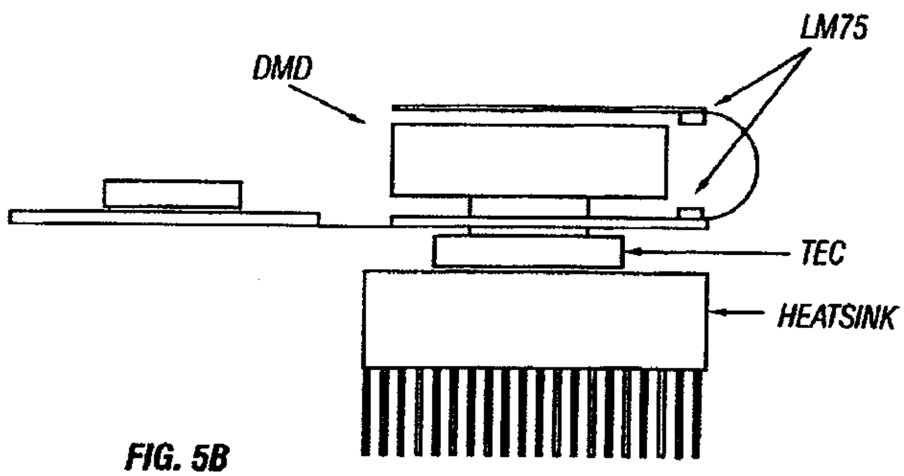


FIG. 5B

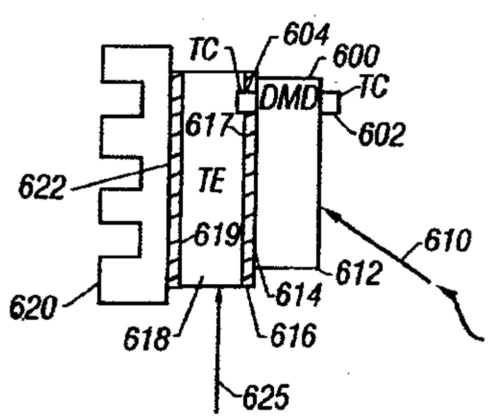


FIG. 6

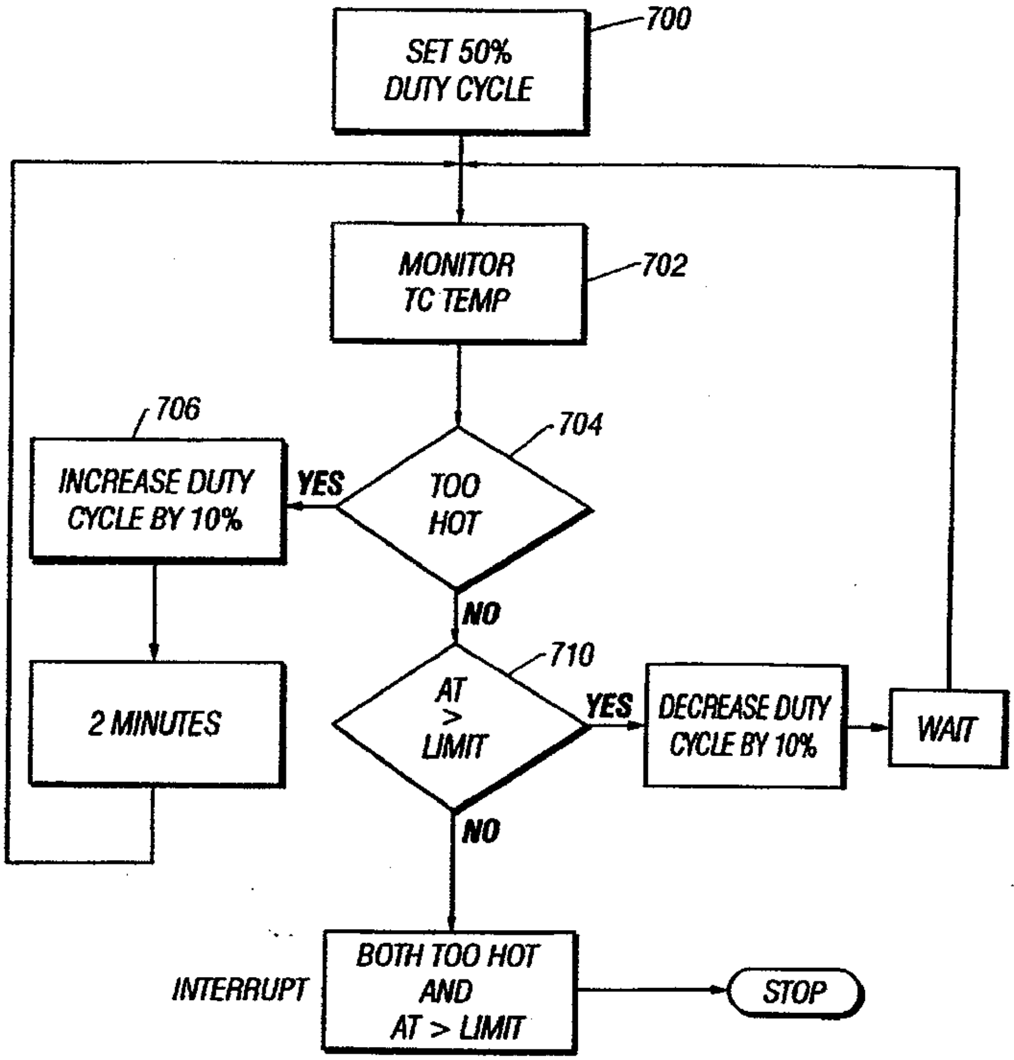

SET 50% DUTY CYCLE
700
MONITOR TC TEMP
702
706
INCREASE DUTY CYCLE BY 10%
YES
TOO HOT
704
NO
2 MINUTES
AT > LIMIT
710
YES
DECREASE DUTY CYCLE BY 10%
WAIT
NO
INTERRUPT
BOTH TOO HOT AND AT > LIMIT
STOP

FIG. 7

PIXEL MIRROR BASED STAGE LIGHTING SYSTEM

This application is a divisional application of U.S. application Ser. No. 09/145,313, filed Aug. 31, 1998, now U.S. Pat. No. 6,208,087 B1.

The present invention describes improvements in a digital mirror stage lighting system. More specifically, the present invention describes techniques which are used to improve operation in the special environment produced by the digital mirror stage lighting system.

BACKGROUND OF THE INVENTION

Stage lighting systems have increased in complexity in recent years. It is desirable to be able to change literally any aspect of the beam projected by a stage light from a remote location. Light & Sound Design, the assignee of the present application, have suggested in other patents and applications that many of these issues could be improved by a stage lighting system that uses an active, x, y addressable element with a digital micromirror device ("DMD") available from Texas Instruments. These devices use an array of controllable mirrors to selectively reflect light in pixel units. These devices have also been called digital mirror, digital light processor (DLP), and other names. Light can be selectively reflected in units of picture elements or pixels. This allows total control of light shape and certain other characteristics. Other devices which are controllable to selectively change characteristics of pixels of light, such as a grating light valve (GLV), can also be used for this purpose.

These concepts are disclosed in several pending patent applications of Light & Sound Design, including Ser. No. 08/854,353, the disclosures of which are herewith incorporated by reference to the extent necessary for proper understanding.

The pixel based light processors, however, themselves produces certain issues, including control issues and cooling issues. The present disclosure describes these and other issues which were found to exist, and describes certain solutions found by the inventors to combat these issues.

SUMMARY OF THE INVENTION

A number of aspects are described according to the present invention, and the following summary explains at least some of these aspects.

A first aspect includes control of the digital mirror device (DMD) and other associated operations. The control is typically completely digital, and many of the operations are carried out entirely mathematically. Therefore, a distributed control with a first control element that carries out mathematical calculations and a second control element which is optimized for control is desirable.

According to an aspect of the embodiment, the second control element is affected out entirely by a programmable gate array, such as a field programmable gate array or similar configurable device. A particularly preferred embodiment reconfigures the device after bulb startup, to avoid noise danger during the bulb startup.

Another aspect is that the digital mirror has loses a certain amount of light. This has required brighter, and hence more power intensive, bulbs. One aspect of this invention is relates to how the operation occurs to keep the heat of the bulb and ignition of the bulb from affecting the other subsystems.

The great amount of heat has produced the need for advanced cooling techniques. However, the digital mirror

has specialized temperature requirements. According to this aspect, a pulse-driven thermoelectric cooler is used and the pulse width to the cooler is changed to change the amount of cooling.

Other aspects of this invention describe the way in which the control element carries out the operation.

BRIEF DESCRIPTION OF THE DRAWINGS

These and other aspects of the invention will now be described in detail with respect to the accompanying drawings, in which:

FIG. 1 shows a block diagram of the electronics of the system;

FIG. 2 shows a flowchart of operation of the system of the supervisor;

FIG. 3 shows a block diagram of the supervisor hardware;

FIG. 4 shows a flowchart of operation of the different processes carried out by the supervisor;

FIGS. 5A and 5B show a service package type cooler and detector device.

FIG. 6 shows the thermoelectric cooler connected to the digital mirror.

FIG. 7 shows a flowchart of the cooling operation carried out by the thermoelectric-device.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

FIG. 1 shows a block diagram of the electronic control system of the preferred embodiment. The TI DLP board **100** is an off the shelf board from Texas Instruments which carries out control of the digital mirror and other pre-defined functions. Associated functions for control of this system are carried out in a field programmable gate array **110** which is preferably of the electronically reconfigurable type. This device is reconfigured into certain logical devices.

The video is controlled by a digital signal processor **150**, in this case, a 320C80. The digital signal processor ("DSP") **150** carries out certain operations under control of the user. DSP **150** also includes two different kinds of slave memory, a flash memory **152** which includes the main program for the DSP **150** and which also includes certain shapes for various controlled lights. Certain information is also stored in synchronous DRAM **154**. On start up, the initial program is transferred from flash memory **152** into the sync DRAM **154** and used to control the digital signal processor **150** and certain other aspects of the electronics. The video device produces an output in the form used by the DLP board **114**.

Serial port **114** is connected to receive data from the controlled console **116**. This data can be sent in any desired serial format and the information is placed on the main data bus **160**. Another serial port **118** is an RS-485 bus driver to form the motor control bus which is described further herein. An IIC port **120** is also formed for other communication operation.

Serial data from the console **116** received via a serial port **114** is input directly to the master digital signal processor ("DSP") **150**, which is preferably a Texas Instruments multimedia video processor ("MVP") model number TMS320C80. DSP **150** uses the information to provide a serial port output which is converted to RS-485 protocol by bus driver serial port driver **118**. The motor control bus is preferably an RS485 bus which controls and communicates with each of the motor subsystems as described in our copending applications.

Each of the motor control subsystems is a separated unit including all of the hardware necessary to control its associated motor and other hardware according to applied commands. The motor control subsystem includes a dedicated control structure. For example, a pan/tilt motor subsystem includes all controlling structure for the motor, and the motor itself. This combination allows a modular operation, precise matching between components, and more accurate calibration.

Master DSP **150** has primary responsibility for controlling operations of the lamp including control of the DMD. This latter operation requires computation of complex operations to provide control information for the DMD. At times, these calculations leave little time for the master DSP **150** to do much else.

A separate supervisor system **199** has primary responsibility for monitoring status of the lamp and making decisions based on that status. Supervisor **199** is also connected to the motor control bus. Supervisor **199** is preferably a microcontroller as described herein. The microcontroller monitors status of the subsystems including the master. The microcontroller can also control the motor control bus when the determined status makes that appear desirable or necessary.

Unlike digital signal processor **150**, however, the microcontroller is a very technically simple device, adapted for watching the bus and other devices, and monitoring for errors. The microcontroller carries out minimal number crunching; its primary function is to protect and diagnose faults. The supervisor also controls various other functions in the system.

The supervisor **199** monitors the output of temperature sensors to monitor and control various temperatures within the system. Supervisor **199** is also connected to the ballast of the lamp to monitor the condition and operation of the ballast. Finally, supervisor **199** receives possible program parameters from flash memory unit **152**.

The operation of the DMD **100** is controlled by master DSP **156** to form any light shape which can be described as a plurality of pixels. A library of possible shapes is stored in image SDRAM **154**.

The serial communications device **114** can also be a dual port RAM with a mailbox. In this case, the information is set into the RAM, and is flagged. The DSP **150** monitors for new data by investigating the flag to determine whether the flag is set. Whenever the flag is set, DSP **150** retrieves the new information from the RAM and appropriately processes it.

As described above, many of the control device structures can be effected using a field programmable gate array **110**. More specifically, any communications port or communications driver, and/or any and all output buffers are preferably formed by reconfiguration of the FPGA. The structures can be described using hardware definition language "HDL" or each of the electronic structures can be configured. Many various canned configurations for FPGAs are well known.

The operation of the system is controlled according to the flowchart of FIG. **2**. The system initially boots at step **200**, which requires transferring the system program from protected flash memory **152** to SDRAM **154**, and beginning operation. At step **202** a global reset is generated. This global reset includes at least the DSP reset vector, and may also force a mechanical reset of all motors.

After the reset at step **202**, the DSP main loop is running. The DSP main loop then sends a command to configure the FPGA **110** at step **204**. The command to configure the FPGA

110 causes the FPGA to form the different logical blocks shown in FIG. **1**. After this, the DSP loop continues shown generically as step **206**. This includes receiving commands from the console, and other housekeeping functions. The normal loop continues until an ignite lamp command is received at step **208**. The lamp is a very high powered lamp, and requires an inductive ballast for operation. The transients produced by the lamp ignition can cause noise transients in the rest of the system.

When the ignite lamp command is received at step **208**, a corresponding command is sent to the supervisor subsystem indicating same at step **210**. The supervisor is a low level system, as described above, which is optimized for control of faults. The supervisor has overall control of many of the reset functions, and at step **212**, initiates a responsive reset. This includes including a break on the motor control bus, holding all the serial com ports in reset, and commanding all boards attached to the supervisor into a hard reset state. The operations in step **212** are done to prepare the system for the inevitable electromagnetic pulse that will be produced by the striking of the bulb. Once all of the above is completed, at step **214**, the supervisor program is halted, and the lamp is ignited. This ignition uses a special subprogram during which no other operations are carried out.

FIG. **2** shows this as halting the supervisor program. After the lamp is ignited at step **214**, the supervisor and all subsystems are released from reset at step **216**. Operation then returns to the normal supervisor loop and the normal DSP loop shown as **230**.

The supervisor, as described above, is optimized for lower level operations and monitoring; compared with the higher level number crunching operations which are carried out by digital signal processor **150**. A block diagram of the supervisor is shown in FIG. **3**. A low level microcontroller such as the 8051 forms the microcontrolling operation. A power supply/regulator **302** supplies microcontroller **300** with its own source of power. This is done to minimize problems from other power supplies.

An external interface to supervisor **199** is provided by the tech port **310**. The tech port **310** is connected to a tech port interface **312** which connects to the microcontroller through a UART. The tech port enables certain operations of the lamp to be individually controlled through the simplified serial interface.

An optional indicator subsystem **320** provides indications of power, data, and error for the operation of the lamp.

Ballast interface **330** connects to the lamp ballast and controls its operation.

The tech port also includes temperature sensor interface **340**. The temperature sensors connect to that interface. An IIC bus driver is also provided as element **350**.

The microcontroller uses associated RAM **360** and real time clock, which is battery powered by battery **362**.

The microcontroller **8051** also includes an RS-485 interface **370** which forms a motor control bus ("MCB").

As described above, many of the features of the supervisor including the interface elements **312/314**, **350** and **370** can be configured out of a field programmable gate array.

The IIC bus **352** connects to a number of different devices, including ROM **354** which includes the yoke serial number, another memory including the head serial number, information indicative of the ballast serial number and the temperature sensor, and the like.

The supervisor tracks bulb life by storing an indication of bulb changing along with the current time stamp, each time

a new bulb is placed into service. Time stamps for other events are also stored. The supervisor also keeps track of certain events, including removal of certain subsystems. It is presumed that these subsystems are serviced when removed. All of this information is stored in a system "registry" in EEPROM **354**.

Certain changes which cannot be automatically detected, such as the time since bulb change, are manually entered into the registry through the tech port **310**.

The information in the registry can be read by the serial device over tech port **310**.

An alterative embodiment allows the information to be commanded to be displayed by the lamp itself as a diagnostic gobo. A lamp display command causes the messages to be converted to fonts and used to control the DMD **100** to display the text error message by shaping the text light output. This allows detecting the contents of the registry without a dedicated display terminal using the existing digital light altering device as a display mechanism. This effectively uses the gobo function of pixel-level addressable device to form a diagnostic function.

The supervisor has a number of operating modes including the initialization mode, locking mode, general task mode, error mode, and technician mode. Each of these modes will be described with reference to the flowchart of FIG. **4**.

The supervisor is initially started in the initialization mode. This mode can use any number of specific internal self tests. The self tests are well known in the art, and are shown generically as step **400**. The internal tests include, but are not limited to, the following:

Internal clock running

External Ram test

IIC bus test

Supervisor temperature sensor test

IIC related bus tests:

Chassis serial number check

Head serial number check

Ballast serial number check

Ballast temperature sensor check

Remote temperature sensor board test

TEC board test

Thermocouples present tests

MCB device tests:

DSP master serial number

Color disk 1 serial number

Color disk 2 serial number

Rear zoom lens serial number

Front zoom lens

Focus lens

Shutter

Custom color

RGB wheel

Pan

Tilt

If the device completes all these tests, control passes to the next mode called the logging mode. The supervisor continually logs certain error messages. Step **402** shows a 2-second interrupt driven output routine. The output routine outputs temperature data and bulb run time every 2 seconds. The error messages can be output every 2 seconds. An alternate technique outputs all data changes as they occur. The output preferably occurs via the tech port **310** although alternately output can be carried out in some other way.

The internal motors of the lamp are also monitored for error messages over the MCB. Each drive module can produce either an error message or a lack of response. Either the error message or the lack of response can be taken as an error.

Certain errors are considered fatal errors and detected at step **406**. For example, over-temperature errors are considered to be fatal, and certain failed movement errors can be considered as fatal. If a fatal error is determined at step **406**, then the bulb is turned off at step **408** to prevent damage, or an erratically-acting lamp unit. An errant light will produce much less visual anomalies if not properly illuminated.

Once the system is placed into the error mode at step **408**, it remains that way until the technician takes control of the fixture using the tech port terminal.

The general task processing loop begins at step **410** with reading temperature sensors and error messages and outputting these values to the tech port as necessary. A number of operations are carried out during these general tasks. The lamp is also carefully monitored.

Each time through the loop, the general tasks shown generally as step **412** are also carried out. These general tasks include:

Incrementing lamp run time counter

Incrementing bulb run time counter if bulb is lit

Entering time and date stamp for lamp turn on

Entering time and date stamp for bulb on and off events

Lamp status after self check

Change of assembly including new serial number after change of assembly

Snapshot of temperatures

Also errors are logged including non-fatal and fatal errors with time and date stamp information for these errors.

At any time during the operation, the technician can take control of the system, shown as step **420**. Normal operation of the lamp is suspended when the technician takes control. This is shown as step **422** which requires a halt when the technician mode is detected at step **420**. In technician mode, the system allows:

Clearing the contents of logged memory by downloading the log

Performing set moves to allow the technician to check lamp operation

Run a built-in light "chase"

View error logs in detail or obtain a status check

Perform diagnostic tests and fault finding using the terminal based on error codes

Downlet preset messages to assemblies via the supervisor to test various operations

In the tech mode, various errors can also be detected and displayed. Each error code has an associated tech string used to describe the error in plain but abbreviated English. This tech string is transmitted through the serial port **312** to the hand held terminal for display. Various tests are described herein.

The following list details certain types and categories of errors that can be detected. This allows the technician to determine if those errors are fatal or non-fatal.

IIC Bus Start Condition Test

Purpose: Checks that the clock and data lines of the IIC bus respond correctly when a start condition is generated on the IIC bus.

Called: Each time an IIC start subroutine is called.

Errors Reported: None; if an error is detected, the IIC bus test routine is called.

IIC Bus Test

Purpose: Checks that the clock and data lines of the IIC bus respond correctly.

Called: At start up and on each read to an IIC device if the lines do not respond correctly during the generation of a start condition (IIC Bus Start Test).

Errors Reported:

Data line lo.

Date line Hi.

Clock line lo.

Clock line Hi.

Clock and data line shorted to each other.

This test looks for the correct operation of the IIC bus clock and data lines. All error codes produced by these tests are FATAL because none of the devices on the bus will be accessible to the supervisor. The supervisor sets the clock and data lines hi and waits for one IIC bus clock cycle. The lines are then tested (testing the port pins, not the output latches of the micro controller). If they are in a hi state, two possible errors can be detected at this point; data line pulled low or clock line pulled low.

IIC Acknowledge Test

Purpose: Checks that the selected IIC device which has been addressed generates an acknowledge signal within 32 IIC bus clock cycles.

Called: After any IIC bus address has been issued to talk to a device.

Errors Reported:

Remote temperature sensing PCG missing.

Ballast temperature sensor missing.

Thermo-electric controller board missing.

Ballast Rom missing.

Yoke Rom missing.

Head Rom missing.

Supervisor temperature sensor missing.

Broken/Missing Thermocouple Test

Purpose: Checks that the selected thermocouple on the remote temperature sensing board is present. If it is missing or broken, it will be flagged as bad in the corresponding status byte for the thermocouple channel.

Called: When each thermocouple is read.

Errors reported:

Thermocouple 1 broken or missing.

Thermocouple 2 broken or missing.

Thermocouple 3 broken or missing.

Thermocouple 4 broken or missing.

Thermocouple 5 broken or missing.

Thermocouple 6 broken or missing.

Thermocouple 7 broken or missing.

Thermocouple 8 broken or missing.

Note: There may be further thermocouples on the TEC board which will be labeled T9 to Tn.

Clock Test

Purpose: Checks that the external time-keeper is running by checking that the seconds register increases by 1 second timed by the micro running a software generated delay.

Called: At start-up.

Errors Reported: Clock error.

Note: It is also possible to test other registers (i.e., minutes or date) for the numbers being in a valid range if desired.

External Ram Test

Purpose: Checks that the external battery backed ram can be read and written.

Called: At start-up.

Errors reported:

Read write error.

Note: It may make sense to have this as part of the clock test routine as a missing clock will probably mean missing ram.

Thermocouple Limit Test

Purpose: Checks that a selected thermocouple channel temperature reading has not exceeded a pre-set limit.

Called: When a thermocouple temperature channel is read.

Errors reported:

Thermocouple 1 exceeded pre-set temperature.

Thermocouple 2 exceeded pre-set temperature.

Thermocouple 3 exceeded pre-set temperature.

Thermocouple 4 exceeded pre-set temperature.

Thermocouple 5 exceeded pre-set temperature.

Thermocouple 6 exceeded pre-set temperature.

Thermocouple 7 exceeded pre-set temperature.

Thermocouple 8 exceeded pre-set temperature.

Thermocouple 1 exceeded upper temperature limit.

Thermocouple 2 exceeded upper temperature limit.

Thermocouple 3 exceeded upper temperature limit.

Thermocouple 4 exceeded upper temperature limit.

Thermocouple 5 exceeded upper temperature limit.

Thermocouple 6 exceeded upper temperature limit.

Thermocouple 7 exceeded upper temperature limit.

Thermocouple 8 exceeded upper temperature limit.

Thermocouple differential pre-set exceeded.

Thermocouple differential upper limit exceeded.

Note:

Two types of limits have been outlined. The pre-set limit warns that temperatures are starting to get dangerously high and may point towards a filter on an air intake becoming blocked and requiring cleaning for example. This type of error would be non-fatal. If the upper temperature level is exceeded it becomes a fatal error and the supervisor would have to take steps to protect the fixture.

The last two differential errors relate to the temperature measurement on the DMD and the thermo-electric cooling. To prevent damage to the device, the temperature difference cannot exceed a certain level.

Digital Temperature Sensor Limit Test

Purpose: Checks that a selected digital temperature sensor reading has not exceeded a pre-set limit.

Called: When a digital temperature sensor is read.

Error reported:

Supervisor exceeded pre-set temperature.

Ballast exceeded pre-set temperature.

Supervisor exceeded upper temperature limit.

Ballast exceeded upper temperature limit

Note:

Two types of limits have been outlined. The pre-set limit warns that temperatures are starting to get dangerously high and may point towards a filter on an air intake becoming blocked and requiring cleaning for example. This type of error would be non-fatal.

If the proper temperature level is exceeded, it becomes a fatal error and the supervisor carries out the processing steps to protect the fixture.

It may well be that no actions will be available to the supervisor to prevent damage other than reducing the power consumption of the fixture.

IIC Serial Number Tests

Purpose: Checks electronic serial number of assemblies on the IIC bus.

Called: During start up.

Errors reported:

Ballast serial number changed.

Remote temperature sensor PCB serial number changed.

TEC serial number changed.

Yoke serial number changed.

Head serial number changed.

Ballast serial number missing/invalid.

Remote temperature sensor PCB serial number missing/invalid.

TEC serial number missing/invalid.

Yoke serial number missing/invalid.

Head serial number missing/invalid.

Note: Two types of errors have been outlined. A missing serial number error may only be required during manufacturing and testing.

MCB Serial Number Checks

Purpose: Checks electronic serial number of assemblies on the MCB bus.

Called: During start up.

Errors reported:

DSP master serial number changed.

Pan assembly serial number changed.

Tilt assembly serial number changed.

Lens 1 assembly serial number changed.

Lens 1 assembly serial number changed.

Focus assembly serial number changed.

RGB assembly serial number changed.

Color disk 1 assembly serial number changed.

Color disk 2 assembly serial number changed.

Customer color disk assembly serial number changed.

Ballast serial number missing/invalid.

Remote temperature sensor PCB serial number missing/invalid.

TEC serial number missing/invalid.

Yoke serial number missing/invalid.

Head serial number missing/invalid.

Note:

Two types of errors have been outlined. A missing serial number error may only be required during manufacturing and testing.

If a module fails to respond, it can be used to detect the absence of a drive module.

MCB Bus Status Byte Tests

Purpose: Checks status bytes of drive modules on the MCB bus.

Called: During monitoring of MCB bus.

Errors reported:

Pan has failed to reset.

Pan drive module over pre-set temperature.

Pan drive module has exceeded maximum temperature.

Pan assembly detected MCB errors.

Tilt has failed to reset.

Tilt drive module over pre-set temperature.

Tilt drive module has exceeded maximum temperature.

Tilt assembly detected MCB errors.

Lens 1 failed to reset.

Lens 1 drive over pre-set temperature.

Lens 1 has exceeded maximum temperature.

Lens 1 lost position.

Lens 1 assembly detected MCB errors.

Lens 2 failed to reset.

Lens 2 drive over pre-set temperature.

Lens 2 has exceeded maximum temperature.

Lens 2 lost position.

Lens 2 assembly detected MCB errors.

Focus failed to reset.

Focus drive over pre-set temperature.

Focus has exceeded maximum temperature.

Focus lost position.

Focus assembly detected MCB errors.

Color 1 failed to reset.

Color 1 drive over pre-set temperature.

Color 1 has exceeded maximum temperature.

Color 1 lost position.

Color 1 assembly detected MCB errors.

Color 2 failed to reset.

Color 2 drive over pre-set temperature.

Color 2 has exceeded maximum temperature.

Color 2 lost position.

Color 2 assembly detected MCB errors.

Custom color failed to reset.

Custom color drive over pre-set temperature.

Custom color has exceeded maximum temperature.

Custom color lost position.

Custom assembly detected MCB errors.

RGB locator failed to reset.

RGB locator drive over pre-set temperature.

RGB locator has exceeded maximum temperature.

RGB locator lost position.

RGB assembly detected MCB errors.

Shutter failed to reset.

Shutter drive over pre-set temperature.

Shutter has exceeded maximum temperature.

Shutter lost position.

Shutter assembly detected MCB errors.

DSP RAM error.

DSP ROM error.

DSP FPGA error.

DLP IIC bus error 1.

DLP IIC bus error 2.

DLP IIC bus error 3.

DLP IIC bus error 4.

ICON data errors (errors detected by DSP master).

MCB bus data errors (framing and parity errors detected by supervisor).

Note: Two types of errors have been outlined. A missing serial number error may only be required during manufacturing and testing.

Definition of errors details how they are detected and if they are fatal or non-fatal. This would require outlining how the operator will perceive the error.

The preferred hand held terminal for the tech port is a micropalm 1200 which has a 320x200 pixel display with gray scale.

The terminal allows different operations to be displayed. FIG. 5 shows a graphical representation of different temperature readings to be shown on the display. Time graphs of errors could also be displayed, e.g., a time access representing the fixture's running time with each column representing, for example, 10 hours of time.

Operation over the tech port allows the following operations.

Diagnostic Mode

Mode accessed by PIN number or password.

Automatic log-in to supervisor with the required terminal details sent to be logged and date and time stamped by the supervisor.

Request a test of the lamp by the supervisor.

View error log to see what errors may have occurred.

View serial number of lamp.

Check bulb burn time.

Run diagnosis routine to fault-find lamp based on the tests.

Save messages in assembly ROMS for use by service department.

Make copies of logs in the lamp for detail examination by service department, i.e., transferring to a computer to be sent by modem.

Manual Mode

Control the lamp using the displayed menu allowing:

Pan

Tilt

Zoom

Shutter

Color

Gobos

Strike

Douse

A test chase can also be done, with the facility to turn individual functions on and off as required.

Service Mode

This allows downloading of memory contents at a service department and is to be done automatically by placing the terminal into a cradle with the data being collected onto a PC.

Each error condition that can be detected will either be FATAL or NON-FATAL and will have a text message string associated with it used to describe the error in English. This section lists all the errors so a decision can be made to the type of error and corresponding message.

2.1.0 Each NON-FATAL error will be prefixed with the string—"!ERROR"

2.2.0 Each FATAL error will be prefixed with the string—"!FATAL ERROR"

2.3.0 IIC bus errors. (All IIC bus errors are probably Fatal) (Fatal)

ET: Message:

2.3.1 Data line shorted to 0V (Fatal)

ET: Message:

2.3.2 Data line shorted to 5V (Fatal)

ET: Message:

2.3.3 Clock line shorted to 0V (Fatal)

ET: Message:

2.3.4 Clock line shorted to 5V (Fatal)

ET: Message:

2.3.5 Clock and data line shorted to each other (Fatal)

ET: Message:

Note: If the IIC bus is faulty no data transactions can occur.

2.4.0 Remote temperature sensing PCB missing (Fatal)

ET: Message:

2.5.0 Ballast temperature sensor missing

ET: Message:

2.6.0 Thermo-electric controller board missing (Fatal)

ET: Message:

Note: The TEC board will control the temperature of the DMD using the thermo-electric controller

2.7.0 Ballast Rom missing

ET: Message:

2.8.0 Yoke Rom missing

ET: Message:

2.9.0 Head Rom missing

ET: Message:

2.10.0 Supervisor temperature sensor missing

ET: Message:

2.11.0 Thermocouple 1 broken or missing

ET: Message:

2.12.0 Thermocouple 2 broken or missing

ET: Message:

2.13.0 Thermocouple 3 broken or missing

ET: Message:

2.14.0 Thermocouple 4 broken or missing

ET: Message:

2.15.0 Thermocouple 5 broken or missing

ET: Message:

2.16.0 Thermocouple 6 broken or missing

ET: Message:

2.17.0 Thermocouple 7 broken or missing

ET: Message:

2.18.0 Thermocouple 8 broken or missing

ET: Message:

2.19.0 Clock error

ET: Message:

2.20.0 Thermocouple 1 exceeded pre-set temperature (Non-Fatal)

ET: Message:

2.21.0 Thermocouple 2 exceeded pre-set temperature (Non-Fatal)

ET: Message:

2.22.0 Thermocouple 3 exceeded pre-set temperature (Non-Fatal)

ET: Message:

2.23.0 Thermocouple 4 exceeded pre-set temperature (Non-Fatal)

ET: Message:

2.24.0 Thermocouple 5 exceeded pre-set temperature (Non-Fatal)

ET: Message:

2.25.0 Thermocouple 6 exceeded pre-set temperature (Non-Fatal)

ET: Message:

2.26.0 Thermocouple 7 exceeded pre-set temperature (Non-Fatal)

ET: Message:

2.27.0 Thermocouple 8 exceeded pre-set temperature (Non-Fatal)

ET: Message:

2.28.0 Thermocouple 1 exceeded upper temperature limit (Fatal)

ET: Message:

2.29.0 Thermocouple 2 exceeded upper temperature limit (Fatal)
ET: Message:
2.30.0 Thermocouple 3 exceeded upper temperature limit (Fatal)
ET: Message:
2.31.0 Thermocouple 4 exceeded upper temperature limit (Fatal)
ET: Message:
2.32.0 Thermocouple 5 exceeded upper temperature limit (Fatal)
ET: Message:
2.33.0 Thermocouple 6 exceeded upper temperature limit (Fatal)
ET: Message:
2.34.0 Thermocouple 7 exceeded upper temperature limit (Fatal)
ET: Message:
2.35.0 Thermocouple 8 exceeded upper temperature limit (Fatal)
ET: Message:
2.36.0 Thermocouple differential pre-set exceeded (Non-Fatal)
ET: Message:
2.37.0 Thermocouple differential upper limit exceeded (Fatal)
ET: Message:
2.38.0 Supervisor exceeded pre-set temperature (Non-Fatal)
ET: Message:
2.39.0 Ballast exceeded pre-set temperature (Non-Fatal)
ET: Message:
2.40.0 Supervisor exceeded upper temperature limit (Fatal)
ET: Message:
Note: The supervisor has a Lithium battery in its external RAM. This will pose a hazard if operated at high temperatures.
2.41.0 Ballast exceeded upper temperature limit (Fatal)
ET: Message:
2.42.0 Ballast serial number changed
ET: Message:
2.43.0 Remote temperature sensor PCB serial number changed
ET: Message:
2.44.0 TEC serial number changed
ET: Message:
2.45.0 Yoke serial number changed
ET: Message:
2.46.0 Head serial number changed
ET: Message:
2.47.0 Ballast serial number missing/invalid
ET: Message:
2.48.0 Remote temperature sensor PCB serial number missing/invalid
2.49.0 TEC serial number missing/invalid
ET: Message:
2.50.0 Yoke serial number missing/invalid
ET: Message:
2.51.0 Head serial number missing/invalid
ET: Message:
2.52.0 DSP master serial number changed
ET: Message:
2.53.0 Pan assembly serial number changed
ET: Message:

2.54.0 Tilt assembly serial number changed
ET: Message:
2.55.0 Lens 1 assembly serial number changed
ET: Message:
2.56.0 Lens 1 assembly serial number changed
ET: Message:
2.57.0 Focus assembly serial number changed
ET: Message:
2.58.0 RGB assembly serial number changed
ET: Message:
2.59.0 Color disk 1 assembly serial number changed
ET: Message:
2.60.0 Color disk 2 assembly serial number changed
ET: Message:
2.61.0 Custom Color disk assembly serial number changed
ET: Message:
2.62.0 Ballast serial number missing/invalid
ET: Message:
2.63.0 Remote temperature sensor PCB serial number missing/invalid
ET: Message:
2.64.0 TEC serial number missing/invalid
ET: Message:
2.65.0 Yoke serial number missing/invalid
ET: Message:
2.66.0 Head serial number missing/invalid
ET: Message:.
2.67.0 Pan has failed to reset
ET: Message:
2.68.0 Pan drive module over pre-set temperature
2.69.0 Pan drive module has exceeded maximum temperature
ET: Message:
2.70.0 Pan assembly detected MCB errors
ET: Message:
2.71.0 Tilt has failed to reset
ET: Message:
2.72.0 Tilt drive module over pre-set temperature
ET: Message:
2.73.0 Tilt drive module has exceeded maximum temperature
ET: Message:
2.74.0 Tilt assembly detected MCB errors
ET: Message:
2.75.0 Lens 1 failed to reset
ET: Message:
2.76.0 Lens 1 drive over pre-set temperature
ET: Message:
2.77.0 Lens 1 has exceeded maximum temperature
ET: Message:
2.78.0 Lens 1 lost position
ET: Message:
2.79.0 Lens 1 assembly detected MCB errors
ET: Message:
2.80.0 Lens 1 corrected loss of position
ET: Message:
2.81.0 Lens 2 failed to reset
ET: Message:
2.82.0 Lens 2 drive over pre-set temperature
ET: Message:
2.83.0 Lens 2 has exceeded maximum temperature
ET: Message:

2.84.0 Lens 2 lost position
ET: Message:
2.85.0 Lens 2 assembly detected MCB errors
ET: Message:
2.86.0 Lens 2 corrected loss of position
ET: Message:
Note: Lens 1 and 2 form the zoom function and share the same mechanical assembly. If one lens has failed to reset or lost position, it may mean that the second lens may collide with the first so both will have to be shut down to prevent damage.
2.87.0 Focus failed to reset
ET: Message:
2.88.0 Focus drive over pre-set temperature
ET: Message:
2.89.0 Focus has exceeded maximum temperature
ET: Message:
2.90.0 Focus lost position
ET: Message:
2.91.0 Focus assembly detected MCB errors
ET: Message:
2.92.0 Focus corrected loss of position
ET: Message:
2.94.0 Color 1 drive over pre-set temperature
ET: Message:
2.95.0 Color 1 has exceeded maximum temperature
ET: Message:
2.96.0 Color 1 lost position
ET: Message:
2.97.0 Color 1 assembly detected MCB errors
ET: Message:
2.98.0 Color 1 corrected loss of position
ET: Message:
2.99.0 Color 2 failed to reset
ET: Message:
2.100.0 Color 2 drive over pre-set temperature
ET: Message:
2.101.0 Color 2 has exceeded maximum temperature
ET: Message:
2.102.0 Color 2 lost position
ET: Message:
2.103.0 Color 2 assembly detected MCB errors
ET: Message:
2.104.0 Color 2 corrected loss of position
ET: Message:
2.105.0 Custom color failed to reset
ET: Message:
2.106.0 Customer color drive over pre-set temperature
ET: Message:
2.107.0 Custom color has exceeded maximum temperature
ET: Message:
2.108.0 Custom color lost position
ET: Message:
2.109.0 Custom assembly detected MCB errors
ET: Message:
2.110.0 Customer corrected loss of position
ET: Message:
2.111.0 RGB locator failed to reset
ET: Message:
2.112.0 RGB locator drive over pre-set temperature
ET: Message:

2.113.0 RGB locator has exceeded maximum temperature
ET: Message:
2.114.0 RGB locator lost position
ET: Message:
2.115.0 RGB assembly detected MCB errors
ET: Message:
2.116.0 Shutter failed to reset
ET: Message:
2.117.0 Shutter drive over pre-set temperature
ET: Message:
2.118.0 Shutter has exceeded maximum temperature
ET: Message:
2.119.0 Shutter lost position
ET: Message:
2.120.0 Shutter assembly detected MCB errors
ET: Message:
2.121.0 Shutter corrected loss of position
ET: Message:
2.122.0 DSP RAM error
ET: Message:
2.123.0 DSP ROM error
ET: Message:
2.124.0 DSP FPGA error
ET: Message:
2.125.0 DLP IIC bus error 1
ET: Message:
2.126.0 DLP IIC bus error 2
ET: Message:
2.127.0 DLP IIC bus error 3
ET: Message:
1.128.0 DLP IIC bus error 4
ET: Message:
2.129.0 ICON data errors (errors detected by DSP master)
ET: Message:
2.130.0 MCB bus data errors (framing and parity errors detected by supervisor)
ET: Message:

Another operation which can be carried out is the diagnostic. gobo. This allows the system to be monitored from the console. Specifically, since the digital mirror device can arrange the shape of light into any desired shape, it can include font shapes. This enables the technician to use the console to focus the image on a screen or any suitable surface. Lamp status can then be seen by the fonts which are prestored and projected by the fixture. This enables checking an entire system relatively quickly.

The same error handling techniques are used, and in this case, the DSP **250** can store font information. This enables projecting the diagnostic gobo information. Of course, certain faults may prevent the diagnostic gobo from operating. For example, if the DSP is not working or if zoom or focus has a problem, then the image is not readable. Failure of pan and tilt may prevent moving the lamp to a position where it is viewable. Moreover, any fatal error will cause the lamp to turn off, hence preventing the image from being seen.

If the lamp can be seen, however, any desired error could be presented. This can include details of custom gobo pallets, lamp status, and bulb life. A non-exclusive list of different errors and their status follows.

Another embodiment is shown in FIG. **6**. Specifically, the DMD **100** has very exacting temperature requirements. The surface of the DMD needs to be kept at 55° C. or less and preferably below 45° C. The inventors believe that the life of the DMD will be improved if it can be kept below 40° C.

The inventors have also recognized the desirability of maintaining the temperature differential, or delta T, across the DMD. The inventors postulate that the delta T should not be greater than 15° C.: i.e. the temperature of the front of the DMD must be within 15° C. of the temperature of the back of the DMD. However, this provides a special cooling challenge for the DMD when cooled from the rear.

The solution, illustrated in FIG. 6, is to use a pulse width modulated thermo-electric cooler. A thermo-electric cooler is a device which pumps cold from one side to the other using the Peltier effect. When current is applied across the thermo-electric device, one side becomes very cold and the other side becomes very hot. This has the effect of essentially pumping the heat from the cold side to the hot side.

The overall structure is shown in FIG. 6. The DMD **600** has thermocouples **602** and **604**, respectively, on the front and rear sides thereof. These thermocouples monitor the surface temperatures of the DMD device.

In operation, incoming light **610** bounces off the front surface **612** of the DMD. A certain amount of this light is converted to heat which hence travels from the front surface.

The back surface **614** of the DMD is coated with thermally conducting paste **616**. This past **616** provides a thermal bond between the back surface of the DMD and the thermoelectric cooler **618**. The cold side **617** of the thermoelectric cooler **618** is pressed against the back surface **614** of the DMD. The hot surface **619** of the thermo-electric cooler **618** is pressed against a cooling heat sink **620** via thermally conductive paste **622**.

In operation, the thermo-electric cooler is energized by an energizing signal **625**. The energizing signal **625**, when active, causes the thermo-electrical cooler to be heated on the hot side **619** and cooled on the cold side **617**. However, this could overcool the back side **614** of the DMD **600** relative to the front **612**. The signal **625** therefore is provided at a specified duty cycle less than 100% to avoid this high thermal gradient.

The DMD is preferably cooled to 30–40° C.

The operation is shown with respect to the flowchart of FIG. 7. As a baseline initial value, a 50% duty cycle is used for the starting operation at step **700**. This value is used when the DMD is operating properly, and during time while the light **610** is being reflected. The temperature of the thermocouple **602** is monitored at step **702**. When step **704** indicates that the front surface of the DMD is getting too hot, the duty cycle is increased by 10% at a time shown at step **706**, and a 2 minute settling time is allowed. Then the duty cycle can be increased by another 10% if necessary. If the delta T from front **612** to back **614** gets too high as detected at step **710**, the duty cycle is reduced to compensate. This allows the overall temperature to be increased some in order to reduce delta T. When both the delta T and temperature get too high, the lamp may need to be doused or dimmed in order to reduce the delta T. This is shown as an interrupt in FIG. 7.

An even further preferred system carries out the temperature sensing and cooling of the device using a service pack device shown in FIGS. 5A and 5B. FIG. 5A shows the unit as laid out in a line. The drive element **800** produces drive signals for providing cooling power to the rear TE device **802** and to the front TE device **804**. Front device **802** and rear device **804** are connected by an electrical connection **806**, which also houses front temp sensor **808** and rear temp sensor **810**. This allows separately cooling the front surface and rear surface, and also separately sensing the temperatures of the first and second surfaces.

Although only a few embodiments have been described in detail above, other embodiments are contemplated by the inventor and are intended to be encompassed within the following claims. In addition, other modifications are contemplated and are also intended to be covered.

What is claimed is:

1. A method of determining service information about a computer controlled lamp, comprising:

determining information related to service of the lamp;

storing said information in a non-volatile memory;

establishing a port that interfaces to said lamp, through which said information can be read; and

allowing said information to be read through said port.

2. A method as in claim **1**, wherein said information includes an indication of bulb life of a bulb within said lamp.

3. A method as in claim **2**, wherein said information includes information indicative of when the bulb is placed into service.

4. A method as in claim **1**, further comprising using an electronic light-shape altering device in said lamp.

5. A method as in claim **4**, wherein said electronic light-shape altering device is used to shape the light into letters which indicate said information.

6. An electronically-controlled lamp, comprising:

a processor;

a light source;

an electronic light-shape altering device, driven by said processor to produce shapes under control of said processor; and

a status detection register, which stores status about said lamp,

wherein said processor controls said altering device according to said status detection register to shape light into letters indicative of information in said status detection register.

7. A device as in claim **6**, wherein said status detection register stores a time of last bulb changing.

8. A device as in claim **6**, wherein said status detection register stores over-temperature errors.

* * * * *

EXHIBIT N

US006549326B2

(12) **United States Patent**
Hunt et al.

(10) **Patent No.:** **US 6,549,326 B2**
(45) **Date of Patent:** **Apr. 15, 2003**

(54) **PIXEL BASED GOBO RECORD CONTROL FORMAT**

(75) Inventors: **Mark A. Hunt**, Derby (GB); **William E. Hewlett**, Burton on Trent (GB); **Ian Clarke**, Walsall (GB)

(73) Assignee: **Light and Sound Design Ltd.**, Birmingham (GB)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/949,155**

(22) Filed: **Sep. 7, 2001**

(65) **Prior Publication Data**

US 2002/0109905 A1 Aug. 15, 2002

Related U.S. Application Data

(62) Division of application No. 09/679,727, filed on Oct. 4, 2000, which is a continuation of application No. 09/495,585, filed on Feb. 1, 2000, now abandoned.

(60) Provisional application No. 60/118,195, filed on Feb. 1, 1999.

(51) **Int. Cl.**⁷ **G02B 26/00**; G06T 1/00; H04N 5/21; G03F 3/08; G05B 11/01

(52) **U.S. Cl.** **359/291**; 382/162; 382/199; 345/418; 348/625; 358/518; 358/1.9; 700/19; 315/292

(58) **Field of Search** 359/291; 382/162, 382/181, 19, 167; 345/418, 419, 431; 348/110, 625; 358/518, 1.9; 700/19; 315/292, 294

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,969,485	A	*	10/1999	Hunt	315/292
6,115,078	A	*	9/2000	Kino	348/625
6,188,933	B1	*	2/2001	Hewlett et al.	700/19
6,226,050	B1	*	5/2001	Lee	348/625
6,393,146	B1	*	5/2002	Doll	382/162
6,400,842	B2	*	6/2002	Fukuda	382/162
6,400,846	B1	*	6/2002	Lin et al.	382/199

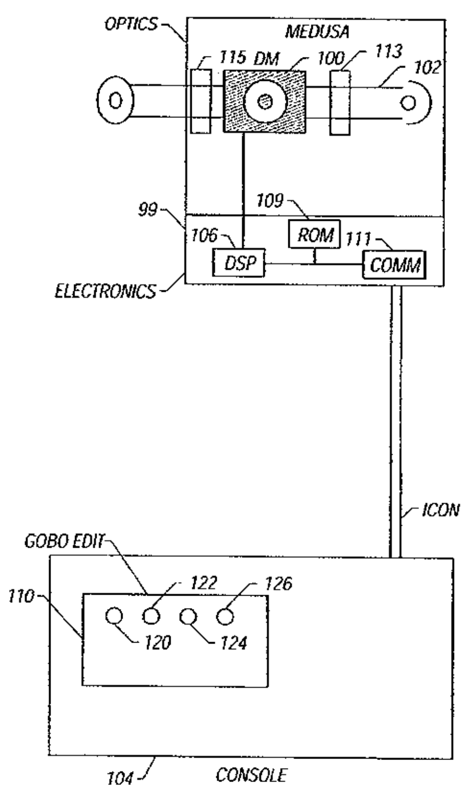
* cited by examiner

Primary Examiner—Loha Ben
(74) *Attorney, Agent, or Firm*—Fish & Richardson P.C.

(57) **ABSTRACT**

Techniques for use in a digital mirror device based luminaire. The techniques include using a filter as a gobo for definition.

24 Claims, 9 Drawing Sheets



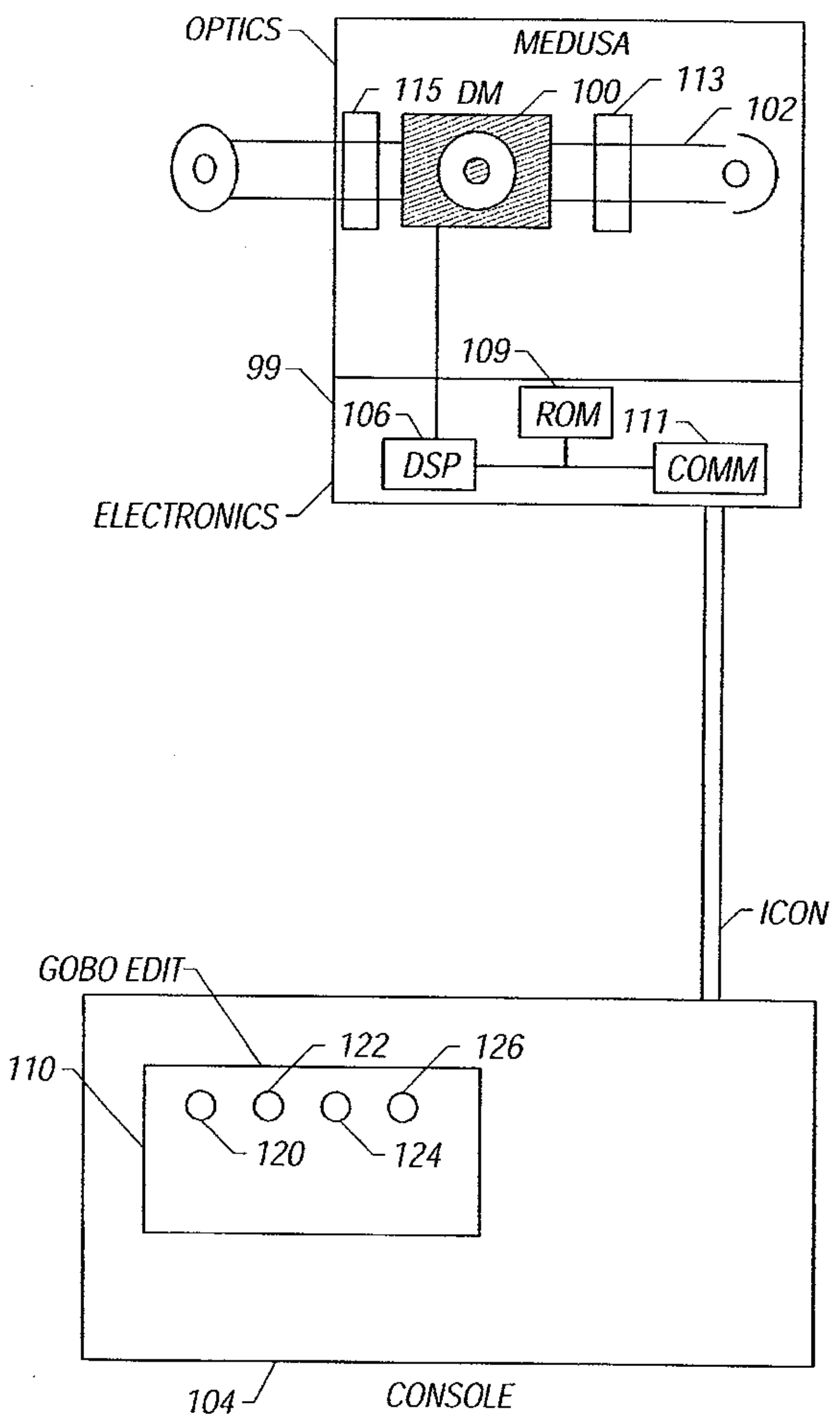

OPTICS
MEDUSA
115 DM
100
113
102
109
99
106
ROM
111
DSP
COMM
ELECTRONICS
ICON
GOBO EDIT
122
126
110
120
124
104
CONSOLE

FIG. 1

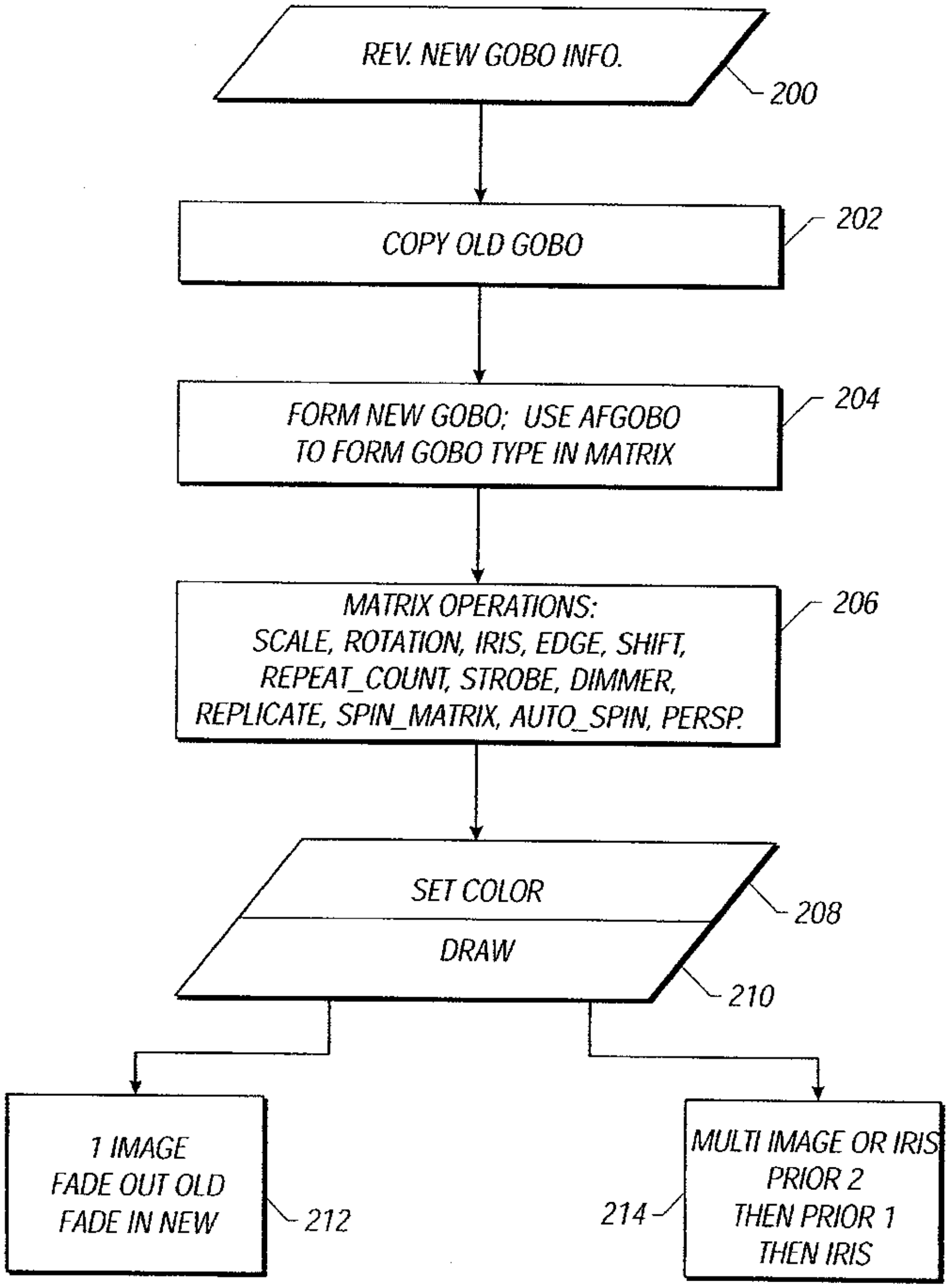
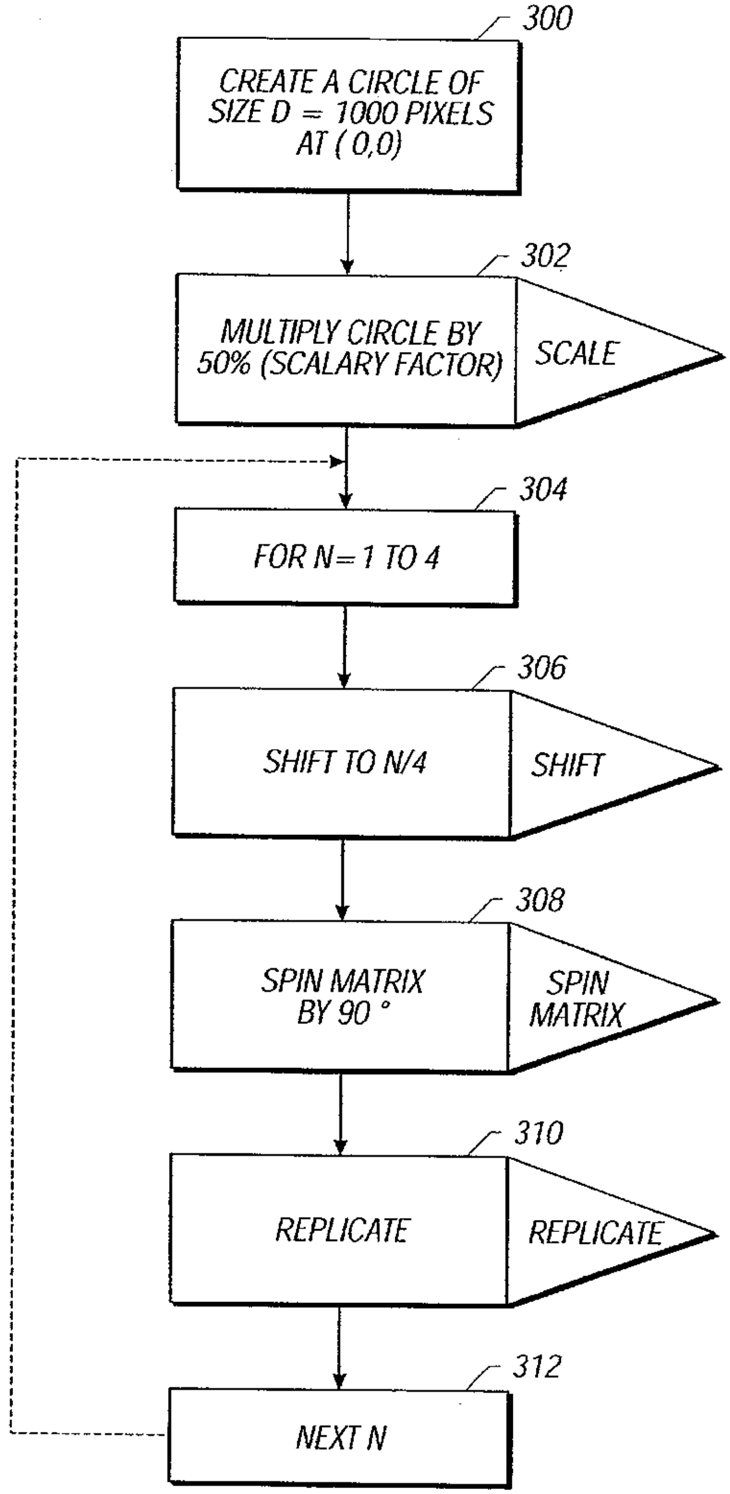

REV. NEW GOBO INFO.
200
COPY OLD GOBO
202
FORM NEW GOBO; USE AFGOBO TO FORM GOBO TYPE IN MATRIX
204
MATRIX OPERATIONS: SCALE, ROTATION, IRIS, EDGE, SHIFT, REPEAT_COUNT, STROBE, DIMMER, REPLICATE, SPIN_MATRIX, AUTO_SPIN, PERSP.
206
SET COLOR
208
DRAW
210
1 IMAGE FADE OUT OLD FADE IN NEW
212
MULTI IMAGE OR IRIS PRIOR 2 THEN PRIOR 1 THEN IRIS
214

FIG. 2



300
CREATE A CIRCLE OF SIZE D = 1000 PIXELS AT (0,0)
302
MULTIPLY CIRCLE BY 50% (SCALARY FACTOR)
SCALE
304
FOR N=1 TO 4
306
SHIFT TO N/4
SHIFT
308
SPIN MATRIX BY 90 °
SPIN MATRIX
310
REPLICATE
REPLICATE
312
NEXT N

FIG. 3

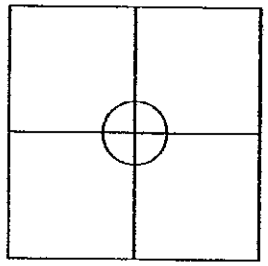


FIG. 4A

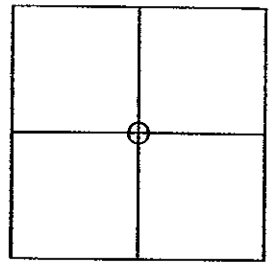


FIG. 4B

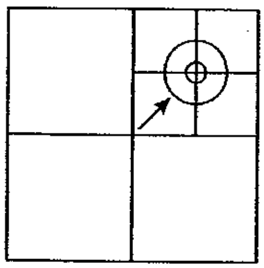


FIG. 4C

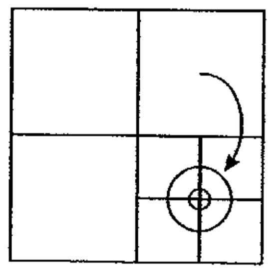


FIG. 4D

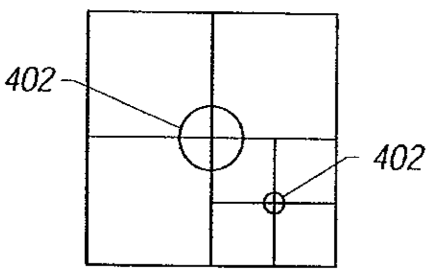


FIG. 4E

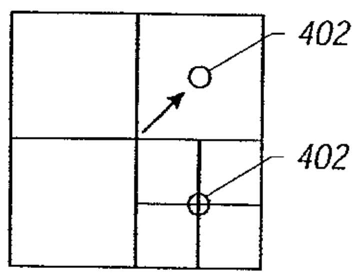


FIG. 4F

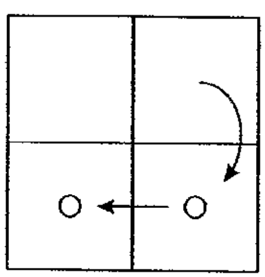


FIG. 4G

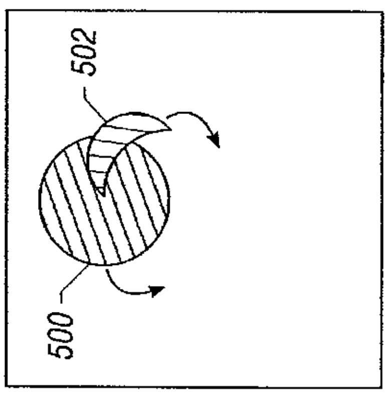


FIG. 5

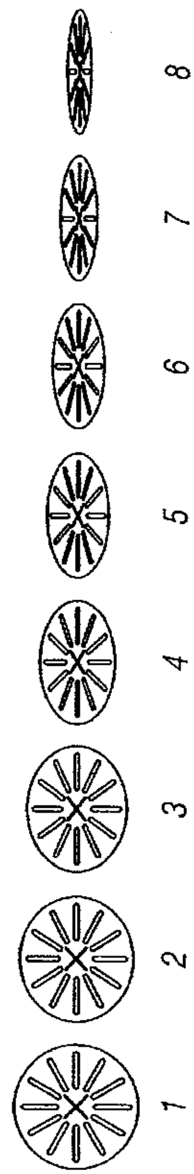


FIG. 6

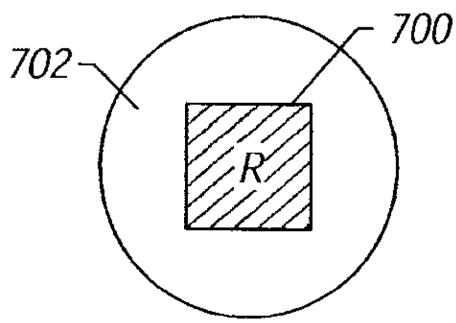


FIG. 7A

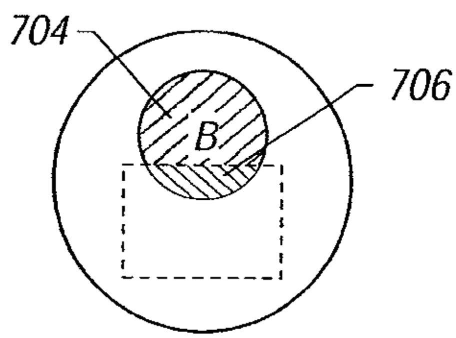


FIG. 7B

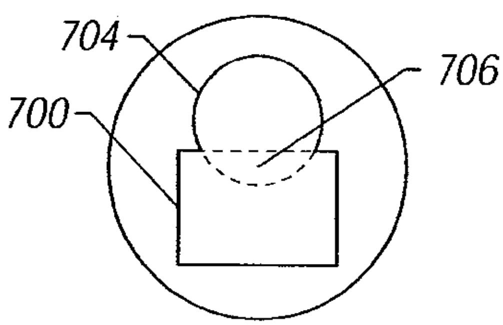
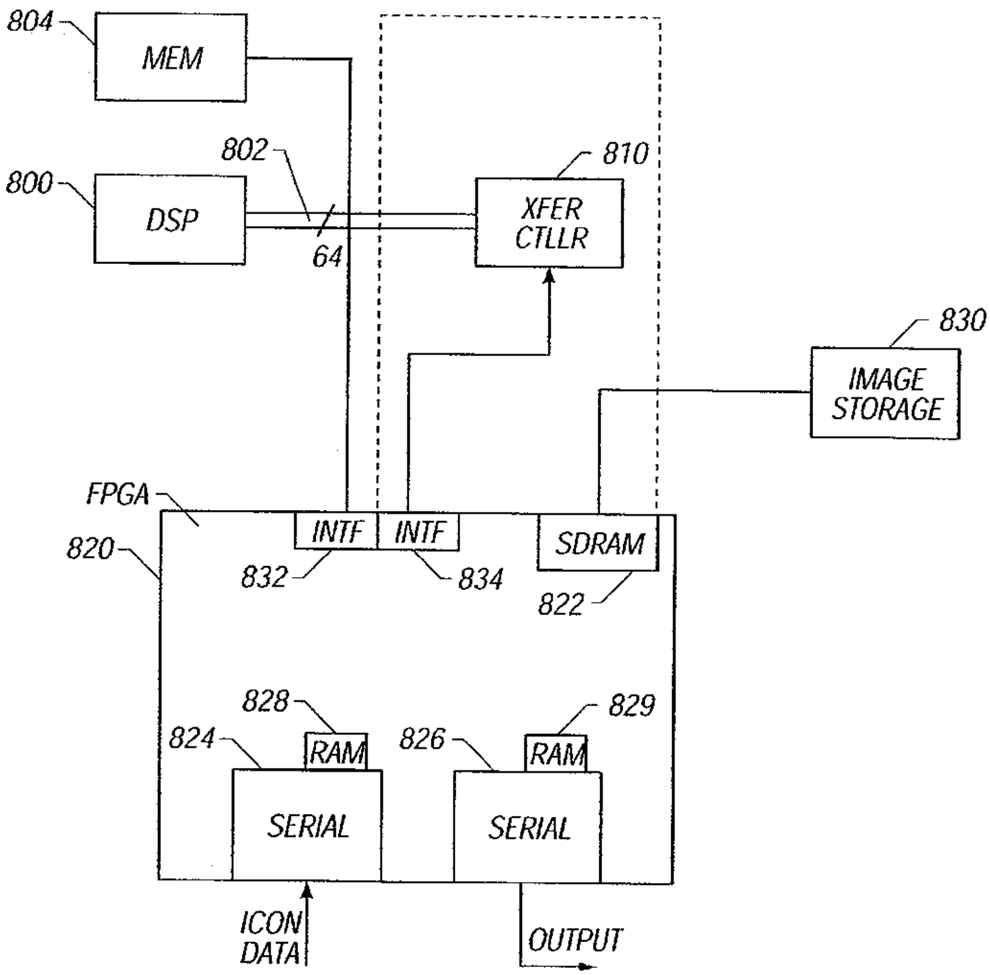
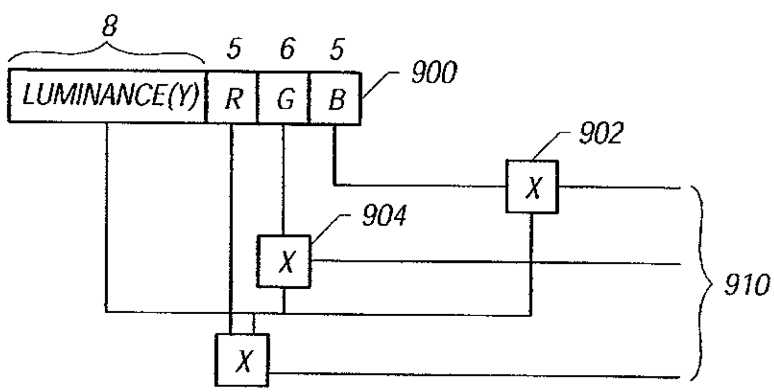


FIG. 7C



804
MEM
802
800
DSP
64
810
XFER
CTLLR
830
IMAGE
STORAGE
FPGA
820
INTF
INTF
SDRAM
832
834
822
828
824
RAM
826
RAM
829
SERIAL
SERIAL
ICON
DATA
OUTPUT

FIG. 8



8
5
6
5
LUMINANCE(Y)
R
G
B
900
902
X
904
X
910
X

FIG. 9

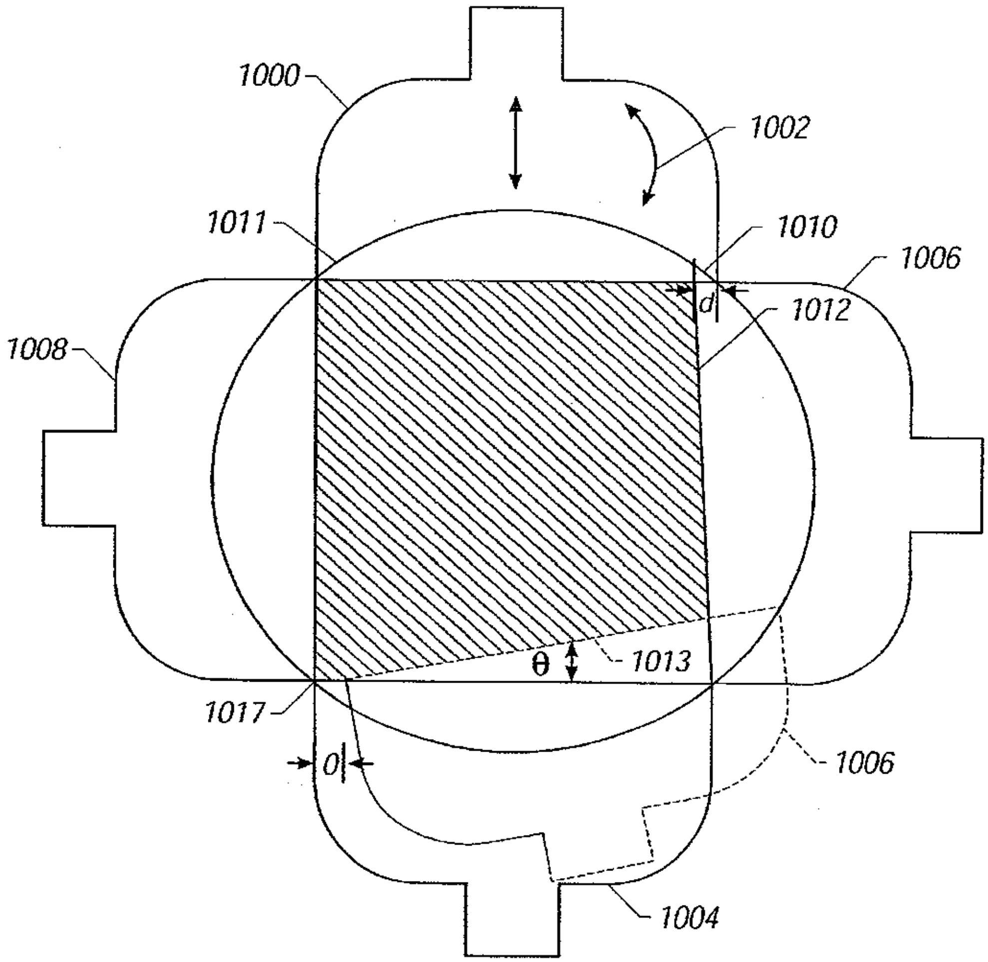
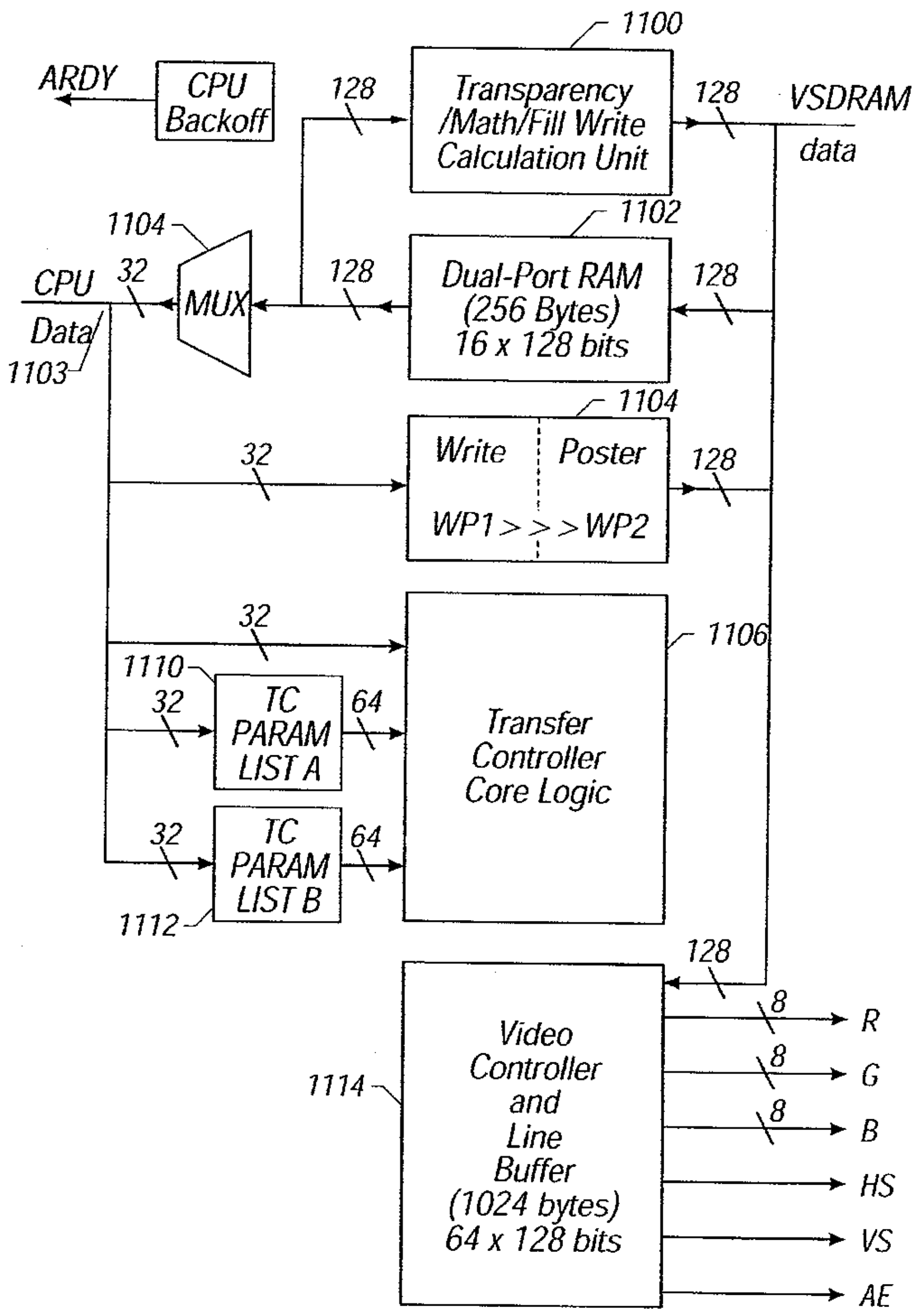


FIG. 10



1100
ARDY
CPU Backoff
128
Transparency /Math/Fill Write Calculation Unit
128
VSDRAM data
1102
1104
CPU
32
MUX
128
Dual-Port RAM (256 Bytes) 16 x 128 bits
128
Data
1103
1104
32
Write
Poster
WP1 > > > WP2
128
32
1106
1110
32
TC PARAM LIST A
64
Transfer Controller Core Logic
32
TC PARAM LIST B
64
1112
128
8
R
8
G
8
B
HS
VS
AE
1114
Video Controller and Line Buffer (1024 bytes) 64 x 128 bits

FIG. 11

PIXEL BASED GOBO RECORD CONTROL FORMAT

CROSS REFERENCE TO RELATED APPLICATIONS

This application is a divisional of U.S. application Ser. No. 09/679,727, filed Oct. 4, 2000, which is a continuation of U.S. application Ser. No. 09/495,585 filed Feb. 1, 2000, now abandoned, which claims the benefit of U.S. provisional application serial No. 60/118,195, filed on Feb. 1, 1999.

FIELD

The present invention relates to a system of controlling light beam pattern ("gobo") shape in a pixilated gobo control system.

BACKGROUND

Commonly assigned patent application Ser. No. 08/854,353, now U.S. Pat. No. 6,188,933, describes a stage lighting system which operates based on computer-provided commands to form special effects. One of those effects is control of the shape of a light pattern that is transmitted by the device. This control is carried out on a pixel-by-pixel basis, hence referred to in this specification as pixilated. The embodiment describes using a digital mirror device, but other x-y controllable devices such as a grating light valve, are also contemplated.

The computer controlled system includes a digital signal processor **106** which is used to create an image command. That image command controls the pixels of the x-y controllable device to shape the light that it is output from the device.

The system described in the above-referenced application allows unparalleled flexibility in selection of gobo shapes and movement. This opens an entirely new science of controlling gobos. The present inventors found that, unexpectedly, even more flexibility is obtained by a special control language for controlling those movements.

SUMMARY

The present disclosure defines aspects that facilitate communicating with an a point controllable device to form special electronic light pattern shapes. More specifically, the present application describes different aspects of communication with an electronic gobo. These aspects include improved processing or improved controls for the gobo.

BRIEF DESCRIPTION OF THE DRAWINGS

These and other aspects of the invention will now be described with reference to the attached drawings, in which:

FIG. **1** shows a block diagram of the basic system operating the embodiment;

FIG. **2** shows a basic flowchart of operation;

FIG. **3** shows a flowchart of forming a replicating circles type gobo;

FIGS. **4**A through **4**G show respective interim results of carrying out the replicating circles operation;

FIG. **5** shows the result of two overlapping gobos rotating in opposite directions;

FIGS. **6(1)** through **6(8)** show a z-axis flipping gobo;

FIGS. **7**A–**7**C show overlapping gobos and then color of overlap;

FIG. **8** shows the black diagram of the system including a transfer controller;

FIG. **9** shows an intensity-sensitive color control elements;

FIG. **10** shows control of a framing shutter; and

FIG. **11** shows a transfer controller made from an FPGA.

DESCRIPTION OF THE PREFERRED EMBODIMENT

FIG. **1** shows a block diagram of the hardware used according to the preferred embodiment. As described above, this system uses a digital mirror device **100**, which has also been called a digital mirror device ("DMD") and a digital light processor device ("DLP"). More generally, any system which allows controlling shape of light on a pixel basis, including a grating light valve, could be used as the light shaper. This light shaper forms the shape of light which is transmitted. FIG. **1** shows the light being transmitted as **102**, and shows the transmitted light. The information for the digital mirror **100** is calculated by a digital signal processor **106**. Information is calculated based on local information stored in the lamp, e.g., in ROM **109**, and also in information which is received from the console **104** over the communication link.

The operation is commanded according to a format.

The preferred data format provides 4 bytes for each of color and gobo control information.

The most significant byte of gobo control data, ("dfGobo") indicates the gobo type. Many different gobo types are possible. Once a type is defined, the gobo formed from that type is represented by a number. That type can be edited using a special gobo editor described herein. The gobo editor allows the information to be modified in new ways, and forms new kinds of images and effects.

The images which are used to form the gobos may have variable and/or moving parts. The operator can control certain aspects of these parts from the console via the gobo control information. The type of gobo controls the gobo editor to allow certain parameters to be edited.

The examples given below are only exemplary of the types of gobo shapes that can be controlled, and the controls that are possible when using those gobo shapes. Of course, other controls of other shapes are possible and predictable based on this disclosure.

First Embodiment

A first embodiment is the control of an annulus, or "ring" gobo. The DMD **100** in FIG. **1** is shown with the ring gobo being formed on the DMD. The ring gobo is type 000A. When the gobo type 0A is enabled, the gobo editor **110** on the console **104** is enabled and the existing gobo encoders **120**, **122**, **124**, and **126** are used. The gobo editor **110** provides the operator with specialized control over the internal and the external diameters of the annulus, using separate controls in the gobo editor.

The gobo editor and control system also provides other capabilities, including the capability of timed moves between different edited parameters. For example, the ring forming the gobo could be controlled to be thicker. The operation could then effect a timed move between these "preset" ring thicknesses. Control like this cannot even be attempted with conventional fixtures.

Another embodiment is a composite gobo with moving parts. These parts can move though any path that is pro-

grammed in the gobo data itself. This is done in response to the variant fields in the gobo control record, again with timing. Multiple parts can be linked to a single control allowing almost unlimited effects.

Another embodiment of this system adapts the effect for an "eye" gobo, where the pupil of the eye changes its position (look left, look right) in response to the control.

Yet another example is a Polygon record which can be used for forming a triangle or some other polygonal shape.

The control can be likened to the slider control under a QuickTime movie window, which allows you to manually move to any point in the movie. However, our controls need not be restricted to timelines.

Even though such moving parts are used, scaling and rotation on the gobo is also possible.

The following type assignments are contemplated:

00_0F=FixedGobo (with no "moving parts")

10_1F=SingleCntrl (with 1 "moving part")

20_2F=DoubleCntrl (with 2 "moving parts")

30_FF=undefined, reserved.

The remaining control record bytes for each type are defined as follows:

Byte	dfGobo2	dfGobo3	dfGobo4	#gobos/type,	total memory
FixedGobo	ID[23:16]	ID[15:8]	ID[7:0]	16 M/type	256 M
SingleCntrl	ID[15:8]	ID[7:0]	control#1	64 k/type	1 M
DoubleCntrl	ID[7:0]	control#2	control#1	256/type	4 k

As can be seen from this example, this use of the control record to carry control values does restrict the number of gobos which can be defined of that type, especially for the 2_control type.

Console Support:

The use of variant part gobos requires no modifications to existing co translate directly to the values of the 4 bytes sent in the communications data packet as follows:

Byte: dfGobo	dfGobo2	dfGobo3	dfGobo4
Enc: TopRight	MidRight	BotRight	BotLeft
FixedGobo:	ID [23:16]	ID [15:8]	ID [7:0]
SingleCntrl:	ID [15:8]	ID [7 0]	control#1
DoubleCntrl:	ID [7:0]	control#2	control#1

These values would be part of a preset gobo, which could be copied as the starting point.

Once these values are set, the third and fourth channels automatically become the inner/outer radius controls. Using two radii allows the annulus to be turned "inside out".

Each control channel's data always has the same meaning within the console. The console treats these values as simply numbers that are passed on. The meanings of those numbers, as interpreted by the lamps change according to the value in dfGobo.

The lamp will always receives all 4 bytes of the gobo data in the same packet. Therefore, a "DoubleCntrl" gobo will always have the correct control values packed along with it.

Hence, the console needs no real modification. If a "soft" console_is used, then name reassignments and/or key reassignments may be desirable.

Timing:

For each data packet, there is an associated "Time" for gobo response. This is conventionally taken as the time allotted to place the new gobo in the light gate. This delay has been caused by motor timing. In this system, variant gobo, the control is more dynamically used. If the non-variant parts of the gobo remain the same, then it is still the same gobo, only with control changes. Then, the time value is interpreted as the time allowed for the control change.

Since different gobo presets (in the console) can reference the same gobo, but with different control settings, this allows easily programmed timed moves between different annuli, etc.

Internal Workings:

When the gobo command data is extracted from the packet at the lamp, the dfGobo byte is inspected first, to see if either dfGobo3 or dfGobo4 are significant in selecting the image. In the case of the "Cntrl" variants, one or both of these bytes is masked out, and the resulting 32-bit number is used to search for a matching gobo image (by Gobo _1D) in the library stored in the lamp's ROM **109**.

If a matching image is found, and the image is not already in use, then the following steps are taken:

1) The image data is copied into RAM, so that its fields may be modified by the control values. This step will be skipped if the image is currently active.

2) The initial control values are then recovered from the data packet, and used to modify certain fields of the image data, according to the control records.

3) The image is drawn on the display device, using the newly-modified fields in the image data.

If the image is already in use, then the RAM copy is not altered. Instead, a time-sliced task is set up to slew from the existing control values to those in the new data packet, in a time determined by the new data packet.

At each vertical retrace of the display, new control values are computed, and steps 2 (using the new control values) and 3 above are repeated, so that the image appears modified with time.

The image data records:

All images stored in the lamp are in a variant record format:

Header:

Length 32 bits, offset to next gobo in list.

Gobo _1D 32 bits, serial number of gobo.

Gobo Records:

Length 32 bits, offset to next record.

Opcode 16 bits, type of object to be drawn.

Data Variant part—data describing object.

_Length 32 bits, offset to next record.

Opcode 16 bits, type of object to be drawn.

Data Variant part—data describing object.

_EndMarker 64 bits, all zeroes—indicates end of gobo data.

+ Next gobo, or End Marker, indicating end of gobo list.

Gobos with controls are exactly the same, except that they contain control records, which describe how the control values are to affect the gobo data. Each control record contains the usual length and Opcode fields, and a field containing the control number (1 or 2).

These are followed by a list of "field modification" records. Each record contains information about the offset (from the start of the gobo data) of the field, the size (8, 16 or 32 bits) of the field, and how its value depends on the control value.

Length	32 bits, offset to next record
Opcode	16 bits = control_record (constant)
CntrlNum	16 bits = 1 or 2 (control number)

-continued

	/* field modification record #1 */
Address	16 bits, offset from start of gobo to affected field.
Flags	16 bits, information about field (size, signed, etc)
Scale	16 bits, scale factor applied to control before use
zPoint	16 bits, added to control value after scaling.
	/* field modification record #2 */
Address	16 bits, offset from start of gobo to affected field.
Flags	16 bits, information about field (size, signed, etc)
Scale	16 bits, scale factor applied to control before use
zPoint	16 bits, added to control value after scaling.

As can be seen, a single control can have almost unlimited effects on the gobo, since ANY values in the data can be modified in any way, and the number of field modification records is almost unlimited.

Note that since the control records are part of the gobo data itself, they can have intimate knowledge of the gobo structure. This makes the hard-coding of field offsets acceptable.

In cases where the power offered by this simple structure is not sufficient, a control record could be defined which contains code to be executed by the processor. This code would be passed parameters, such as the address of the gobo data, and the value of the control being adjusted.

Example Records.

The Annulus record has the following format:

Length	32 bits
Opcode	16 bits, = type_annulus
Pad	16 bits, unused
Centre_x	16 bits, x coordinate of centre
Centre_y	16 bits, y coordinate of centre
OuterRad	16 bits, outside radius (the radii get swapped when drawn if their values are in the wrong order)
InnerRad	16 bits, inside radius

It can be seen from this that it is easy to "target" one of the radius parameters from a control record. Use of two control records, each with one of the radii as a target, would provide full control aver the annulus shape.

Note that if the center point coordinates are modified, the annulus will move around the display area, independent of any other drawing elements in the same gobo's data.

The Polygon record for a triangle has this format:

Length	32 bits
Opcode	16 bits, = type_polygon
Pad	16 bits, vertex count = 3
Centre_x	16 bits, x coordinate of vertex
Centre_y	16 bits, y coordinate of vertex
Centre_x	16 bits, x coordinate of vertex
Centre_y	16 bits, y coordinate of vertex
Centre_x	16 bits, x coordinate of vertex
Centre_y	16 bits, y coordinate of vertex

It is easy to modify any of the vertex coordinates, producing distortion of the triangle.

The gobo data can contain commands to modify the drawing environment, by rotation, scaling, offset, and color control, the power of the control records is limitless.

Second Embodiment

This second embodiment provides further detail about implementation once the gobo information is received.

Gobo information is, at times, being continuously calculated by DSP **106**. The flowchart of FIG. **2** shows the handling operation that is carried out when new gobo information is received.

At step **200**, the system receives new gobo information. In the preferred embodiment, this is done by using a communications device **111** in the lamp **99**. The communications device is a mailbox which indicates when new mail is received. Hence, the new gobo information is received at step **200** by determining that new mail has been received.

At step **202**, the system copies the old gobo and switches pointers. The operation continues using the old gobo until the draw routine is called later on.

At step **204**, the new information is used to form a new gobo. The system uses a defined gobo ("dfGobo") as discussed previously which has a defined matrix. The type dfGobo is used to read the contents from the memory **109** and thereby form a default image. That default image is formed in a matrix. For example, in the case of an annulus, a default size annulus can be formed at position 0,0 in the matrix. An example of forming filled balls is provided herein.

Step **206** represents calls to subroutines. The default gobo is in the matrix, but the power of this system is its ability to very easily change the characteristics of that default gobo. In this embodiment, the characteristics are changed by changing the characteristics of the matrix and hence, shifting that default gobo in different ways. The matrix operations, which are described in further detail herein, include scaling the gobo, rotation, iris, edge, strobe, and dimmer. Other matrix operations are possible. Each of these matrix operations takes the default gobo, and does something to it.

For example, scale changes the size of the default gobo. Rotation rotates the default gobo by a certain amount.

Iris simulates an iris operation by choosing an area of interest, typically circular, and erasing everything outside that area of interest. This is very easily done in the matrix, since it simply defines a portion in the matrix where all black is written.

Edge effects carry out certain effects on the edge such as softening the edge. This determines a predetermined thickness, which is translated to a predetermined number of pixels, and carries out a predetermined operation on the number of pixels. For example, for a 50% edge softening, every other pixel can be turned off. The strobe is in effect that allows all pixels to be turned on and off at a predetermined frequency, i.e., 3 to 10 times a second. The dimmer allows the image to be made dimmer by turning off some of the pixels at predetermined times.

The replicate command forms another default gobo, to allow two different gobos to be handled by the same record. This will be shown with reference to the exemplary third embodiment showing balls. Each of those gobos are then handled as the same unit and the entirety of the gobos can be, for example, rotated. The result of step **206** and all of these subroutines that are called is that the matrix includes information about the bits to be mapped to the digital mirror **100**.

At step **208**, the system then obtains the color of the gobos from the control record discussed previously. This gobo color is used to set the appropriate color changing circuitry **113** and **115** in the lamp **99**. Note that the color changing

circuitry is shown both before and after the digital mirror **100**. It should be understood that either of those color changing circuits could be used by itself.

At step **210**, the system calls the draw routine in which the matrix is mapped to the digital mirror. This is done in different ways depending on the number of images being used. Step **212** shows the draw routine for a single image being used as the gobo. In that case, the old gobo, now copied as shown in step **202**, is faded out while the new gobo newly calculated is faded in. Pointers are again changed so that the system points to the new gobo. Hence, this has the effect of automatically fading out the old gobo and fading in the new gobo.

Step **214** schematically shows the draw routine for a system with multiple images for an iris. In that system, one of the gobos is given priority over the other. If one is brighter than the other, then that one is automatically given priority. The one with priority 2, the lower priority 1, is written first. Then the higher priority gobo is written. Finally, the iris is written which is essentially drawing black around the edges of the screen defined by the iris. Note that unlike a conventional iris, this iris can take on many different shapes. The iris can take on not just a circular shape, but also an elliptical shape, a rectangular shape, or a polygonal shape. In addition, the iris can rotate when it is non-circular so that for the example of a square iris, the edges of the square can actually rotate.

Returning to step **206**, in the case of a replicate, there are multiple gobos in the matrix. This allows the option of spinning the entire matrix, shown as thin matrix.

An example will now be described with reference to the case of repeating circles. At step **200**, the new gobo information is received indicating a circle. This is followed by the other steps of **202** where the old gobo is copied, and **204** where the new gobo is formed. The specific operation forms a new gobo at step **300** by creating a circle of size diameter equals 1000 pixels at origin 00. This default circle is automatically created. FIG. **4A** shows the default gobo which is created, a default size circle at 00. It is assumed for purposes of this operation that all of the circles will be the same size.

At step **302**, the circle is scaled by multiplying the entire circle by an appropriate scaling factor. Here, for simplicity, we are assuming a scaling factor of 50% to create a smaller circle. The result is shown in FIG. **4B**. A gobo half the size of the gobo of FIG. **4A** is still at the origin. This is actually the scale of the subroutine as shown in the right portion of step **302**. Next, since there will be four repeated gobos in this example, a four-loop is formed to form each of the gobos at step **304**. Each of the gobos is shifted in position by calling the matrix operator shift. In this example, the gobo is shifted to a quadrant to the upper right of the origin. This position is referred to as □ over 4 in the FIG. **3** flowchart and results in the gobo being shifted to the center portion of the top right quadrant as shown in FIG. **4C**. This is again easily accomplished within the matrix by moving the appropriate values. At step **308**, the matrix is spun by 90 degrees in order to put the gobo in the next quadrant as shown in FIG. **4D** in preparation for the new gobo being formed into the same quadrant. Now the system is ready for the next gobo, thereby calling the replicate command which quite easily creates another default gobo circle and scales it. The four-loop is then continued at step **312**.

The replicate process is shown in FIG. **4E** where a new gobo **402** is formed in addition to the existing gobo **400**. The system then passes again through the four-loop, with the results being shown in the following figures. In FIG. **4F**, the new gobo **402** is again moved to the upper right quadrant (step **306**). In FIG. **4G**, the matrix is again rotated to leave room for a new gobo in the upper right quadrant. This continues until the end of the four-loop. Hence, this allows each of the gobos to be formed.

Since all of this is done in matrix operation, it is easily programmable into the digital signal processor. While the above has given the example of a circle, it should be understood that this scaling and moving operation can be carried out for anything. The polygons, circles, annulus, and any other shape is easily scaled.

The same operation can be carried out with the multiple parameter gobos. For example, for the case of a ring, the variable takes the form annulus (inner R, outer R, x and y). This defines the annulus and turns of the inner radius, the outer radius, and x and y offsets from the origin. Again, as shown in step 3, the annulus is first written into the matrix as a default size, and then appropriately scaled and shifted. In terms of the previously described control, the ring gobo has two controls: control 1 and control 2 defined the inner and outer radius.

Each of these operations is also automatically carried out by the command repeat count which allows easily forming the multiple position gobo of FIGS. **4A–4G**. The variable auto spin defines a continuous spin operation. The spin operation commands the digital signal processor to continuously spin the entire matrix by a certain amount each time.

One particularly interesting feature available from the digital mirror device is the ability to use multiple gobos which can operate totally separately from one another raises the ability to have different gobos spinning in different directions. When the gobos overlap, the processor can also calculate relative brightness of the two gobos. In addition, one gobo can be brighter than the other. This raises the possibility of a system such as shown in FIGS. **5A–5C**. Two gobos are shown spinning in opposite directions: the circle gobo **500** is spinning the counterclockwise direction, while the half moon gobo **502** is spinning in the clockwise direction. At the overlap, the half moon gobo which is brighter than the circle gobo, is visible over the circle gobo. Such effects were simply not possible with previous systems. Any matrix operation is possible, and only a few of those matrix operations have been described herein.

A final matrix operation to be described is the perspective transformation. This defines rotation of the gobo in the Z axis and hence allows adding depth and perspective to the gobo. For each gobo for which rotation is desired, a calculation is preferably made in advance as to what the gobo will look like during the Z axis transformation. For example, when the gobo is flipping in the Z axis, the top goes back and looks smaller while the front comes forward and looks larger. FIGS. **6(1)–6(8)** show the varying stages of the gobo flipping. In FIG. **6(8)**, the gobo has its edge toward the user. This is shown in FIG. **6(8)** as a very thin line, e.g., three pixels wide, although the gobo could be zero thickness at this point. Automatic algorithms are available for such Z axis transformation, or alternatively a specific Z axis transformation can be drawn and digitized automatically to enable a custom look.

Third Embodiment

The gobo record format described above can have two gobos therein. These two gobos can be gobo planes, which can be used to project one image superimposed over another image in a predefined way. For example, a first image can be

a pattern that emits light, e.g., a standard gobo. The second image can be totally transparent, or can have holes through which the first image can be seen.

Analog gobos often project light through two gobos. The light is then projected through the intersection between the two gobos. Effectively, this takes an AND function between the gobos. Light will only be passed in places where both gobos are open.

In the present system, any function between two images can be projected as an overall gobo shape. The system can, e.g., project an "or" operation between the two images. Moreover, the two images can be projected in separate colors. The operation could be carried out in software.

A first gobo shown in FIG. 7A is a square gobo. For purposes of this example, the square gobo is projected in red ("R"), forming a first red lighted portion. The exterior non-projected portion **702** is black.

FIG. 7B shows the second gobo to be combined with the first gobo. The second gobo is an off-center circle **704** to be projected in blue ("B"). The AND between these two gobos would transmit only the intersection between the two gobos, shown by the hatched portion **706**. Moreover, this portion could only be transmitted in the additive or subtractive combination between the two colors, red and blue.

The present system defines the two images as conceptually being separate planes. This enables transmitting the "or", or any other combination, between the two images. Both the first image **700** and the second image **704** are displayed. Moreover, the intersection portion of the image **706** can be made in any desired color, either the color of either, the color of the subtractive combination, or a totally different color. While this system describes an "or" operation, it also encompasses any combination between the gobos: e.g., exclusive or, Schmitt-triggered (hysteresis-induced combination) AND/OR, or others.

The gobo operation is also simplified and made more efficient by using a transfer controller as described herein.

FIG. **8** shows the basic block diagram of this embodiment. The Digital Signal Processor (DSP) **800** effectively functions as the central processing unit. A DSP for this embodiment is the TI TMS 320C80. This has a 64-bit bus **802**. Memory **804** is attached to the bus **802**. The memory **804** effectively forms a working portion. A transfer controller **810** is provided and allows increased speed. The transfer controller can take control of the bus and can carry out certain functions. One such function is a direct memory access. This allows moving information from the program memory **804** to a desired location.

The transfer controller receives information about the data to be moved, including the start location of the data, the number of bytes of the data, and the end location of the data. The destination and operation is also specified by the data **809**. The transfer controller **810** then takes the data directly from the memory **804**, processes it, and returns it to the memory or to the DLP without DSP intervention. The CPU can then therefore instruct the transfer controller to take some action and then can itself do something else.

Hardware block **820** also connects to the bus **802**. This is preferably formed from a Field Programmable Gate Array (FPGA). The FPGA can be configured into logical blocks as shown. The DSP also sends commands that reconfigure the FPGA as needed. The FPGA can be reconfigured to form fast Synchronous Dynamic Random Access Memory (SDRAM) shown as **822**.

DSP **800** can be a TI TMS 320C80. This device includes an associated transfer controller which is a combined

memory controller and DMA (direct memory access) machine. It handles the movement of data and instructions within the system as required by the master processor, parallel processors, video controller, and external devices.

The transfer controller performs the following data-movement and memory-control functions:

MP and ADSP instruction-cache fills
MP data-cache fills and dirty-block write-back
MP and ADSP packet transfers (PTs)
Externally initiated packet transfers (XPTs)
VC packet transfers (VCPTs)
MP and ADSP direct external accesses (DEAs)
VC shift-register-transfer (SRTs)
DRAM refresh
External bus requests

Operations are performed on the cache sub-block as requested by the processors' internal cache controllers. DEA operations transfer off-chip data directly to or from processor registers. Packet transfers are the main data transfer operations and provide an extremely flexible method for moving multidimensional blocks of data (packets) between on-chip and/or off-chip memory.

Key features of this specific transfer controller include:

Crossbar interface,
64-bit data path,
Single-cycle access,
External memory interface,
4G-byte address range dynamically configurable memory cycles,
Bus size of 8, 16, 32, or 64 bits,
Selectable memory page size,
Selectable row/column address multiplexing,
Selectable cycle timing,
Big or little endian operation Cache, VRAM, and refresh controller,
Programmable refresh rate,
VRAM block-write support,
Independent source and destination addressing,
Autonomous address generation based on packet transfer parameters;
Data can be read and written at different rates
Numerous data merging and spreading functions can be performed during transfers; and
Intelligent request prioritization

Hence, the transfer controller allows definition of the limits of the message/data. Then, the information can be automatically handled. The transfer controller can also generate a table of end points, carry out direct-memory access, and manipulate the data while transferring the data.

The SDRAM **822** can be used as fast-image memory, and can be connected, for example, to an image storage memory **830**. The FPGA can also be configured to include serial interfaces **824, 826** with their associated RAM **828, 829** respectively. Other hardware components also can be configured by the FPGA.

Since the FPGA can be reconfigured under control of the digital signal processor **800**, the FPGA can be reconfigured dynamically to set an appropriate amount of SDRAM **822**. For example, if a larger image or image processing area is necessary, the FPGA can be reconfigured to make more of its area into image memory. If a smaller image is desired, less of the FPGA can be made into SDRAM, allowing more

of the FGPA for other hardware functions. Moreover, the interfaces **832**, **834** can be dynamically reconfigured. For example, the baud rate can be changed, bus width can be reconfigured, and the like.

The video controller and line buffer **1114** can also be formed from the field-programmable gate array.

The serial receiver **824** receives the lamp data from the controller, as described in U.S. Pat. No. 5,969,485. The serial driver **826** produces a serial output that can drive, for example, an RS422 bus that runs the motors.

The C80 DSP includes the transfer controller as a part thereof.

An alternative embodiment uses a different DSP. The functions of the transfer controller are then replicated in the FPGA, as desired. For example, an alternative possible DSP is the C6201 which uses the Very Large Instruction Word "VLIW" architecture. This system can use, for example, 128-bit instructions. However, since this is connected to the 32-bit data bus, a transfer controller could be highly advantageous. This would enable the equivalent of direct memory access from the memory. FIG. **11** shows the gate array schematic of this alternate embodiment in which the transfer controller is part of the FPGA.

A second embodiment of the gate array logic, as arranged according to the present system, is shown in FIG. **11**. This gate array logic is formed in the field-programmable gate array **820** to carry out many of the functions described herein. Block **1100** corresponds to a transparency device which calculates values associated with transparency.

Block **1102** is a dual-port RAM which receives the VLIW at one port thereof, and outputs that value to a multiplexer **1104**, which outputs it as a 32 bit signal used by the CPU/DSP.

Transfer controller **1106** has the functionality discussed above. It is controlled directly by the CPU data received on line **1105**. The transfer controller can have two lists of parameters, each 64 bits in width. These values are received on the list receivers **1110**, **1112**.

Another issue noted by the current inventors is the size of images. If possible, it is desirable to avoid using uncompressed images. For example, one simple form image to manipulate is a bitmap, also known as a ".bmp" type image. The bitmap represents each pixel of the image by a number of bits, e.g., for an 8-bit 3-primary color image, each pixel would require 24 bits. This can, unfortunately, use incredible amounts of storage. However, since the bit map has a 1-to-1 correspondence with the image, it can be relatively easy to manipulate the bit map. For example, a matrix representing the bitmap can be easily manipulated, e.g., rotated. The image form can be compressed, e.g., to a GIF or JPEG image. This compressed image, however, loses the one-to-one correspondence and hence cannot be directly processed as easily.

One aspect of the present system is to store the image as a compressed image, and most preferably as polygons. The software package, Adobe Streamline (TM), breaks a bitmap into multiple polygons. The polygons can then be defined as vectors. An additional advantage is that the vectors can be easily processed by the DSP. The DSP **800** then builds the image from the vectors. Since the image is defined as vectors, it can be easily handled via matrix arithmetic. Using Adobe Streamline, for example, an 800 kilobyte bit map can be compressed to a 30 kilobyte vector image.

Another improvement of the present system is the control of the gobo using filters.

In an analog gobo system, a filter can be used to blur the image representing the gobo, for example. Many different kinds of filters are used. For example, some filters randomly distort the image. Other filters affect the image in different ways. The blurring can be carried out as an electronic filter. A preferred user interface defines the filter as a separate gobo that is multiplied, e.g., AND ed, or OR ed with the first gobo.

More generally, a filter can be used to alter the image in some way, e.g., scale the image, decay the image, or the like. The blur can be used to make the image apparently out of focus in some locations.

The filter uses a second gobo that simulates the effect of an analog filter. For example, one operation simulates the optical effect of the glass that forms the filter in an analog gobo. That glass is used to make a model that emulates the optical properties of the glass. Those optical properties are then manipulated through the matrix representing the gobo, thereby effecting a digital representation of the filter. In one aspect, the filter is considered as a separate gobo which is OR ed with the second gobo. In this case, the dual gobo definition described above can be used. Alternatively, the filter can simply be added to the gobo-defining matrix.

This definition has the advantage that it avoids defining a totally separate control. The filters are each defined as one specific gobo. A user manual which defines gobos is used. This manual has filters added to it. This avoids the need for a separator user manual of filters.

Another aspect defined by the present system is gobos that load and execute code. Some images cannot be described in terms of control. For example, images may be defined as some random input. Some images progress with time and maintain no record of their previous state. These images can be defined in terms of code and in terms of a progression from one time to another. Hence, the gobos that load and execute code define a gobo that includes an associated area to hold static values.

A gobo is requested. The code and variables that are associated with that gobo are copied into RAM. The variables are initially at a preset state. The code that is in the gobo portion is executed, using the portions in the variables. The variables are modified at each pass through the portion.

Yet another feature of this system is intensity control over aspects of the image defining the gobo and dimming of the image defined thereby. Returning to the example of a bit map with 24-bit color, such a system would include 8 bits of red, 8 bits of green, and 8 bits of blue. It can be desirable to fade the image while keeping the color constant with intensity change.

One system uses an experimental technique i.e. that is one that relies on experimentation, to determine how to fade in order to maintain constant color. A look-up table is formed between the constant color and the look up table. In this way value B_x, G_x, B_x represents color 1 at intensity X. Ry, Gy, By represent the color at intensity y.

Another system directly maps the bits to color by defining the map as chrominance using techniques from color television. For example, this takes the bits, and converts the values indicating image to color or chrominance (C) and image luminance (Y) of the image. The conversion between RGB and Y/C is well known. The values of Y and C which correspond to the chrominance and luminance are then stored. The image gobo can then be dimmed by reducing the Y, while keeping C the same. If desired, the Y/C can be converted back to RGB after dimming. The dimming however, may change the "look" of the color being projected. This system allows the color to be changed based on intensity.

Another system allows reducing the number of bits for a bitmap. Say, as an example, that it is desired to use a total

of 8 bits to represent each pixel of the image. This could then be apportioned between the desired bits with red having 3 bits, green having 3 bits, and blue having 2 bits. This limits the amount of information in any of these colors. Since there are only 2 bits for blue, there are only four levels of blue that can be selected. This is often insufficient.

In this system, therefore, the bits are compressed by assuming that each two adjacent lines have exactly the same values. Hence, each two lines get the same color value (but can have different intensity values). Now in a system as described above, two lines of red can have 5 bits, two lines of green can have 6 bits, and two lines of blue can also have 5 bits. This provides an appropriate dynamic range for color at the expense of losing half the resolution for color.

Moreover, this has an additional advantage in that it allows 5 bits for grey scale in such a system.

A possible problem with such a system, however, as described above, is that the information would not necessarily be aligned on byte boundaries. It could, therefore, be necessary to take the whole image, manipulate it, and then put the whole image back.

The basic system is shown in FIG. **9**. The luminance Y is an 8-bit representation of the brightness level of the image. The hue is then divided into dual-line multiple bits. Each value is used for two lines each.

Dimming in such a system is carried out as shown in FIG. **9**. For example, the blue bits **900** are multiplied in a hardware multiplier **902** by the luminance. Similarly, the green is multiplied in a second hardware multiplier **904** by the same luminance value. This controls the relative levels of red, green, and blue that are output on the RGB lines **910**.

The multipliers that are used are very simple, since they simply multiply 8 bits by 3 bits. Therefore, a relatively simple in structure hardware multiplier can be used for this function.

This provides red, green, and blue color without loss of data and with substantially perfect fading.

An additional feature described herein is a framing shutter gobo. A basic framing shutter is shown in FIG. **10**. FIG. **10** shows the circular spot of the beam, and the analog shutter, often called a LECO. Each analog shutter **1000** can be moved in and out in the direction of the arrows shown. Each shutter can also be moved in an angular direction, shown by the arrow **1002**. There are a total of four shutters, which, in combination, enable framing the beam to a desired shape. For example, the shutter **1004** can be moved to the position shown in dotted lines as **1006**. When this happens, the effective image that is passed becomes as shown in hatched lines in FIG. **10**. Another possibility is that the shutter can be tilted to put a notch or nose into the window around the image.

According to this system, a record is formed for a gobo defining a framing shutter. The framing shutter gobo allows control of multiple values including the positions of the four framing shutter edges **1000**, **1004**, **1006**, and **1008**. Each framing shutter is defined in terms of its value d, corresponding to the distance between one edge **1010** of the framing shutter and the edge **1011** of the original spot. In this system, the value d is shown representing the right-hand edge of the framing shutter. Another selectable value is θ, which defines the angle that the front blade **1013** of the framing shutter makes relative to perfect horizontal or vertical. Yet another parameter which can be selected is offset O which represents the distance between the framing shutter edge **1010** and the ideal edge portion **1017**. Other values can alternatively be specified. By controlling all these values, the Medusa shutter can in effect simulate any desired framing shutter by using an electronic gobo.

A number of different special gobos are defined according to the present system. Each of these gobos is defined according to the record format described above.

These include:

Oscilloscope. This enables simulating the output value of an oscilloscope as the gobo. For example, any value that can be displayed on the oscilloscope could be used as a gobo with a finite width. This could include sine waves, square waves, straight waves, sawtooth waves, and the like.

Other variable gobos include vertical lines, moire lines, laser dots, radial lines, concentric circles, geometric spiral, bar code, moon phases, flowers and rotating flowers, a diamond tiling within a shape, kaleidoscope, tunnel vision, and others.

Animated gobos correspond to those which execute codes described above. Some examples of these include, for example, self-animating random clouds; self-animating random reflections; self-animating random flames, fireworks; randomly moving shapes such as honeycombs, crosswords, or undulations; foam; random flying shapes.

Although only a few embodiments have been described in detail above, those having ordinary skill in the art certainly understand that modifications are possible.

What is claimed is:

1. A processing system for a digitally-controllable light passing element, comprising:

a memory, storing a digital file that represents a shape of light to be passed;

a digital signal processor, which carries out, in operation, mathematical operations on said digital file;

a transfer controller element, separate from said digital signal processor, which receives information about data to be moved, including start location of the data, and other information which enables the device to determine the data, and which obtains the data directly from the memory, processes it according to the requests, and returns the information to the memory, without intervention of the digital signal processor;

and uses said information to modify said digital file; and

a hardware block, which receives and interfaces commands from a remote controller.

2. A device as in claim **1** wherein said hardware block is formed from a configured FPGA.

3. A device as in claim **2** wherein said digital signal processor configures the FPGA.

4. A device as in claim **2** wherein said FPGA is formed into dynamic RAM blocks.

5. A devices as in claim **2** wherein said FPGA is configured to form input and output ports.

6. A device as in claim **2** wherein said transfer controller is formed from said FPGA.

7. A device as in claim **2** wherein said transfer controller is separate from the FPGA.

8. A method of controlling a digital gobo, comprising:

forming an image representing a gobo from a plurality of polygons; and

using said image to control an electronic element to shape an output light.

9. A method as in claim **8** wherein said polygons are vectorized polygons.

10. A method as in claim **8** further comprising using said image to control a digital mirror device to display light according to information in said image.

11. A method as in claim **8** further comprising filtering said image using a filter.

12. A method of projecting light, comprising:

forming an image which will be used as a gobo for said light, to shape an outer edge of said light;

compressing said image;

storing the compressed version of said image; and

using said compressed version of said image to control an electronic element to shape said light.

13. A method as in claim 12 wherein said compressed image is compressed using vectors.

14. A method as in claim 13 wherein the vectorized image is processed using matrix arithmetic.

15. A method as in claim 13 wherein said compressing comprises dividing the image into multiple polygons, and defining said polygons in terms of vectors.

16. A method of storing information for controlling a gobo, comprising:

storing a first image representing a gobo shape;

storing a second image, representing a filter used to distort the gobo shape and using said first and second images to control an electronic device to display an image.

17. A method as in claim 16 wherein said filter defines an object which is mathematically applied to said gobo.

18. A method as in claim 16 wherein said filter comprises a scale of the image or a decay of the image.

19. A method as in claim 16 wherein said filter comprises a blur of the image.

20. A method as in claim 16 wherein said filter comprises a gobo that simulates an effect of an analog filter.

21. A method as in claim 20 wherein said effect of the analog filter is an effect of optical properties of specified glass.

22. A method of controlling a digital light controlling element, comprising:

storing an image representation in a memory, said image representation indicating a basic gobo;

modifying said image representation using a second gobo acting as a filter to form a modified image; and

using the modified image to control the digital light controlling element, to display light.

23. A method as in claim 22 wherein said filter includes a specified gobo.

24. A method as in claim 22 wherein said gobos hold static values enabling execution of code.

* * * * *

EXHIBIT O

US006597132B2

(12) **United States Patent**
Hunt et al.

(10) **Patent No.: US 6,597,132 B2**
(45) **Date of Patent: *Jul. 22, 2003**

(54) **STAGE LIGHTING LAMP UNIT AND STAGE LIGHTING SYSTEM INCLUDING SUCH UNIT**

(75) Inventors: **Mark A. Hunt**, Derby (GB); **Keith J. Owen**, Moseley (GB); **Michael D. Hughes**, Wolverhampton (GB)

(73) Assignee: **Light and Sound Design Ltd.**, Birmingham (GB)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

This patent is subject to a terminal disclaimer.

(21) Appl. No.: **10/007,008**

(22) Filed: **Dec. 4, 2001**

(65) **Prior Publication Data**

US 2002/0070689 A1 Jun. 13, 2002

Related U.S. Application Data

(63) Continuation of application No. 09/313,418, filed on May 17, 1999, now Pat. No. 6,326,741, which is a continuation of application No. 08/994,036, filed on Dec. 18, 1997, now Pat. No. 5,921,659, which is a division of application No. 08/576,211, filed on Dec. 21, 1995, now Pat. No. 5,788,365, which is a continuation of application No. 08/077,877, filed on Jun. 18, 1993, now Pat. No. 5,502,627.

(30) **Foreign Application Priority Data**

Sep. 25, 1992 (GB) 9220303
Sep. 25, 1992 (GB) 9220309
Apr. 20, 1993 (GB) 9308071

(51) **Int. Cl.**[7] **H05B 37/00**
(52) **U.S. Cl.** 315/316; 315/312; 362/301; 362/233
(58) **Field of Search** 315/316, 312, 315/294, 292, 318, 319; 362/301, 233, 293

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,392,187	A	*	7/1983	Bornhorst	362/233
4,716,344	A	*	12/1987	Newell et al.	315/294
4,980,806	A	*	12/1990	Taylor et al.	315/316
5,005,162	A	*	4/1991	Mitsumori et al.	369/112.29
5,010,459	A	*	4/1991	Taylor et al.	315/292
5,045,983	A		9/1991	Shields	367/293
5,414,328	A	*	5/1995	Hunt et al.	315/292
5,502,627	A	*	3/1996	Hunt et al.	362/286
5,588,021	A	*	12/1996	Hunt et al.	370/246
5,769,531	A	*	6/1998	Hunt et al.	362/233
5,788,365	A	*	8/1998	Hunt et al.	362/301
5,921,659	A	*	7/1999	Hunt et al.	362/233
6,326,741	B1	*	12/2001	Hunt et al.	315/316

FOREIGN PATENT DOCUMENTS

EP	0 495 305	7/1992
FR	2 643 134	8/1990
GB	2 209 229	5/1989
GB	2 231 138	11/1990
WO	89 054 21	6/1989

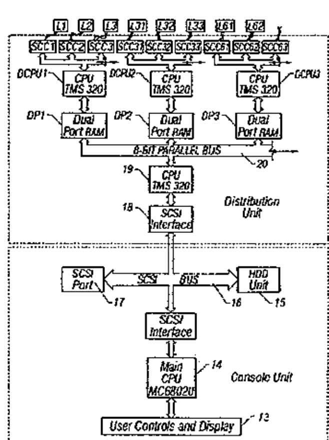
* cited by examiner

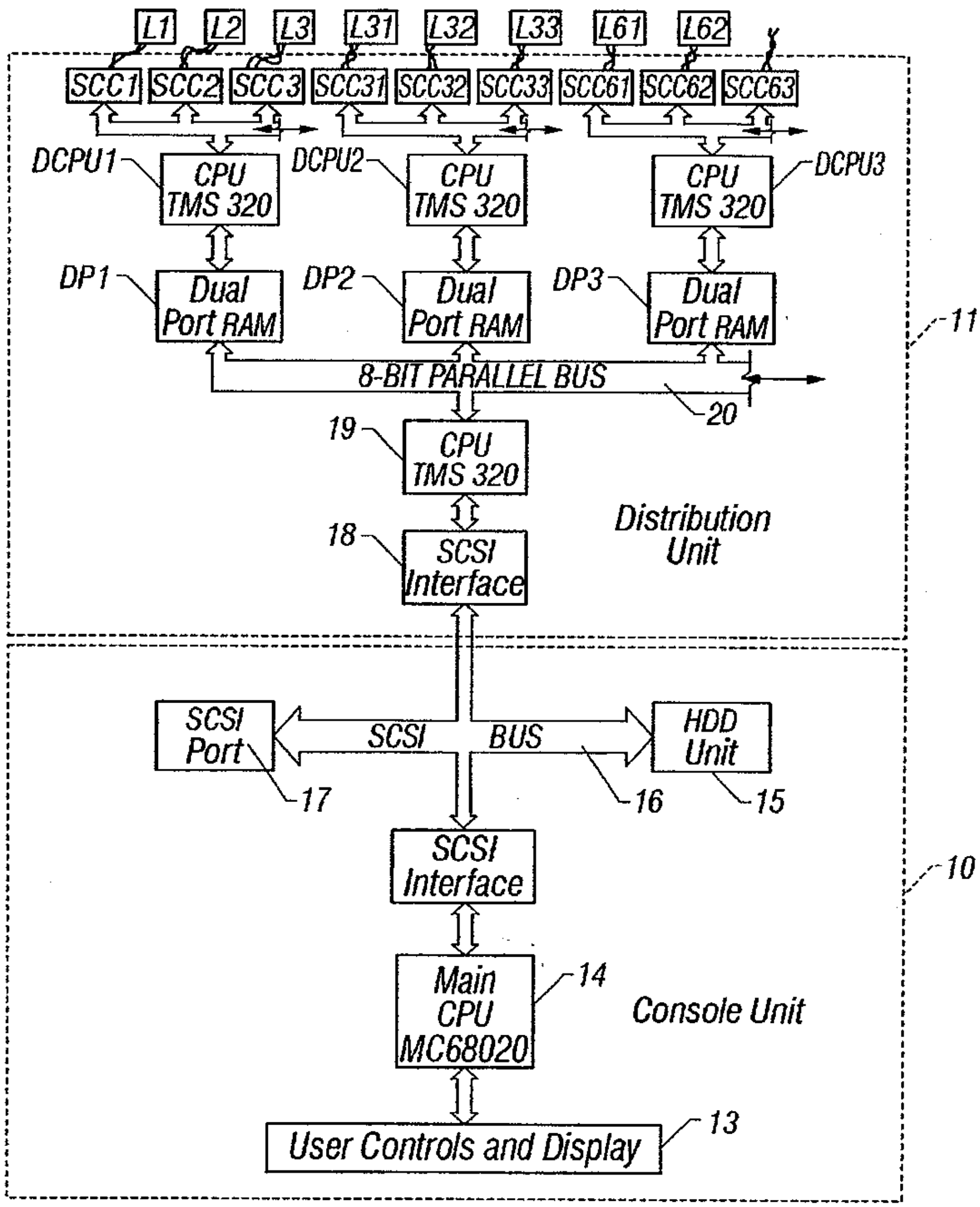
Primary Examiner—Don Wong
Assistant Examiner—Ephrem Alemu
(74) *Attorney, Agent, or Firm*—Fish & Richardson P.C.

(57) **ABSTRACT**

A stage lighting lamp unit includes a processor for receiving control data from a remote console. Beam orientation data for the lamp unit is passed to the lamp in the form of the x, y and z co-ordinates of a point in space through which the beam is to pass. The processor divides the required lamp travel into a number of stages dependent on execution duration data sent with the position data, and calculates, for each stage, a new value for pan and tilt angles for the lamp. These values are passed to pan and tilt controlling co-processors which control servo-motors for pan and tilt operation. The lamp unit also incorporates a rotatable shutter for interrupting the lamp beam when required. The shutters of all the lamps in a system can be instructed from the remote console to open and close in synchronism, thereby providing a stroboscopic effect.

52 Claims, 10 Drawing Sheets





L1
L2
L3
L31
L32
L33
L61
L62
SCC1
SCC2
SCC3
SCC31
SCC32
SCC33
SCC61
SCC62
SCC63
DCPU1
DCPU2
DCPU3
CPU TMS 320
CPU TMS 320
CPU TMS 320
DP1
DP2
DP3
Dual Port RAM
Dual Port RAM
Dual Port RAM
11
8-BIT PARALLEL BUS
20
19
CPU TMS 320
18
SCSI Interface
Distribution Unit
SCSI Port
SCSI BUS
HDD Unit
17
16
15
SCSI Interface
10
Main CPU MC68020
14
Console Unit
User Controls and Display
13

FIG. 1

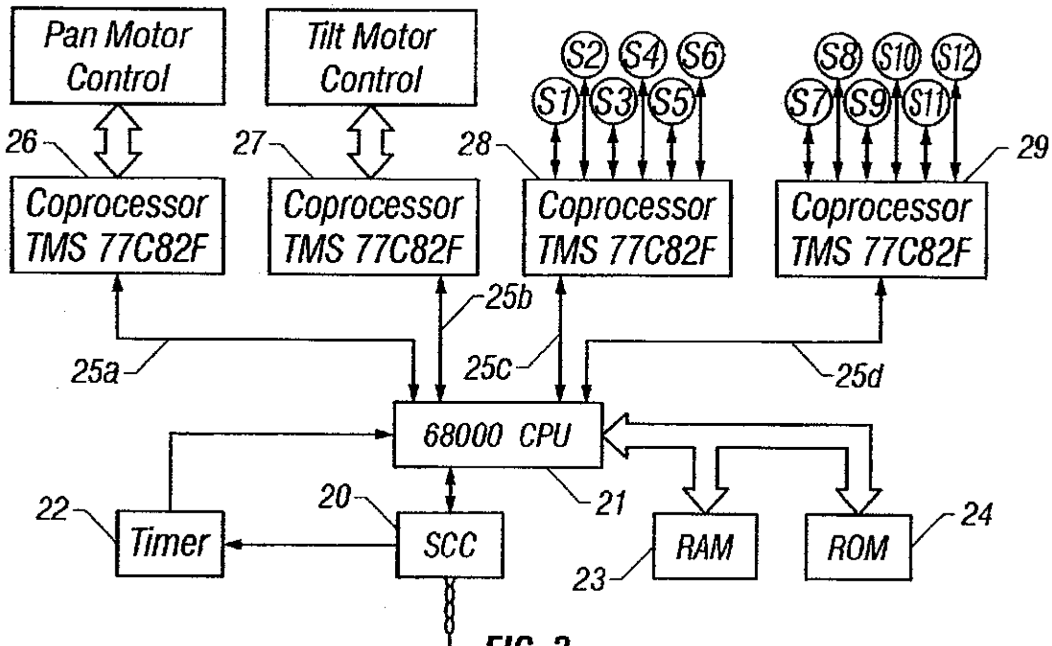

Pan Motor Control
Tilt Motor Control
S2
S4
S6
S8
S10
S12
S1
S3
S5
S7
S9
S11
26
27
28
29
Coprocessor TMS 77C82F
Coprocessor TMS 77C82F
Coprocessor TMS 77C82F
Coprocessor TMS 77C82F
25b
25a
25c
25d
68000 CPU
22
20
21
24
Timer
SCC
RAM
ROM
23

FIG. 2

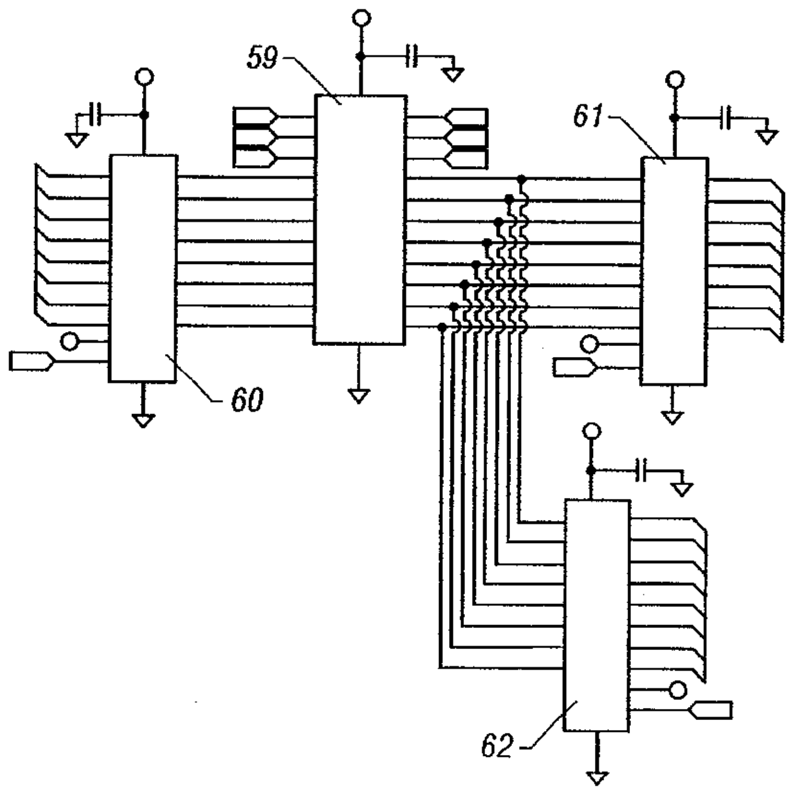

59
61
60
62

FIG. 7

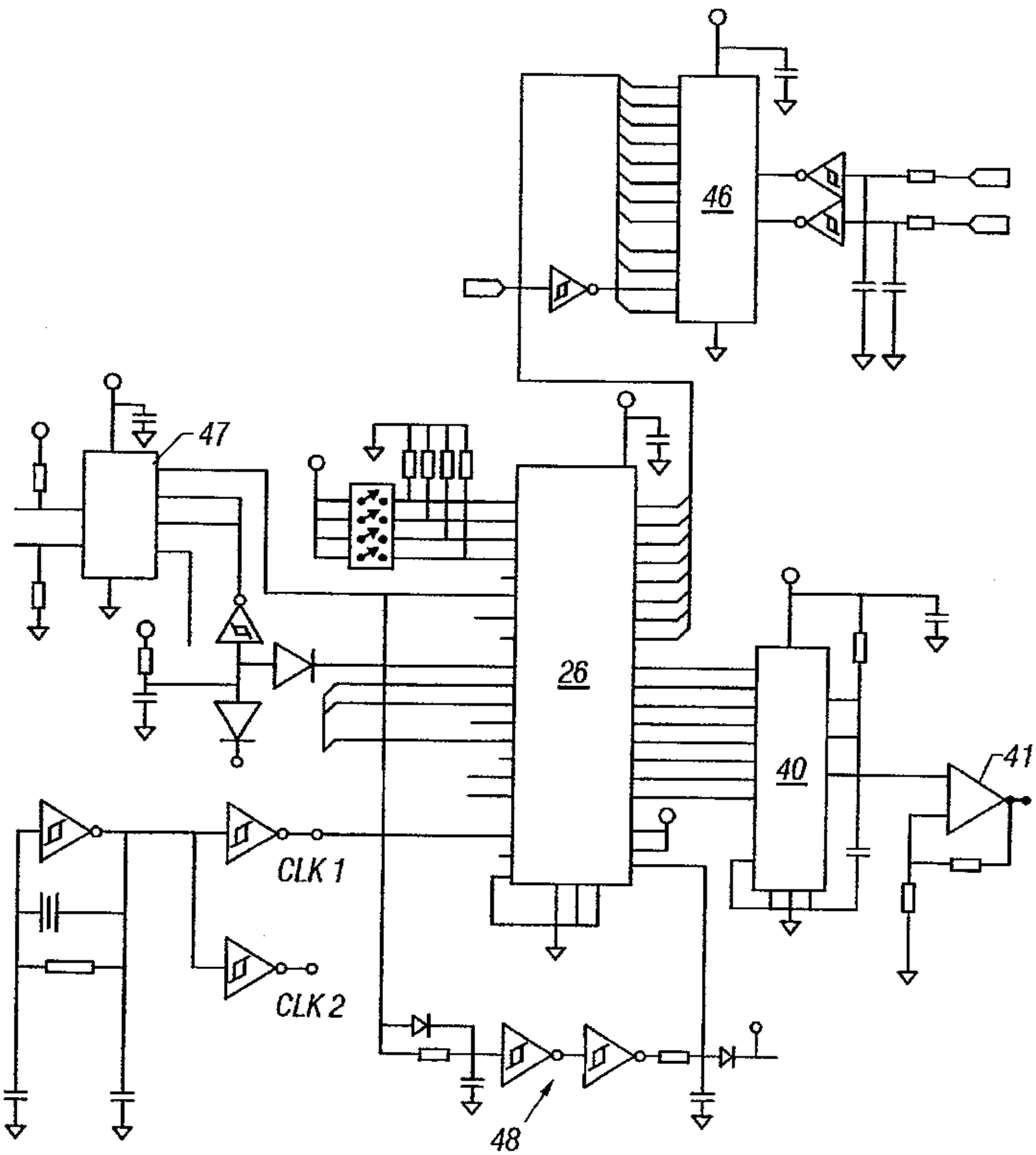


FIG. 3

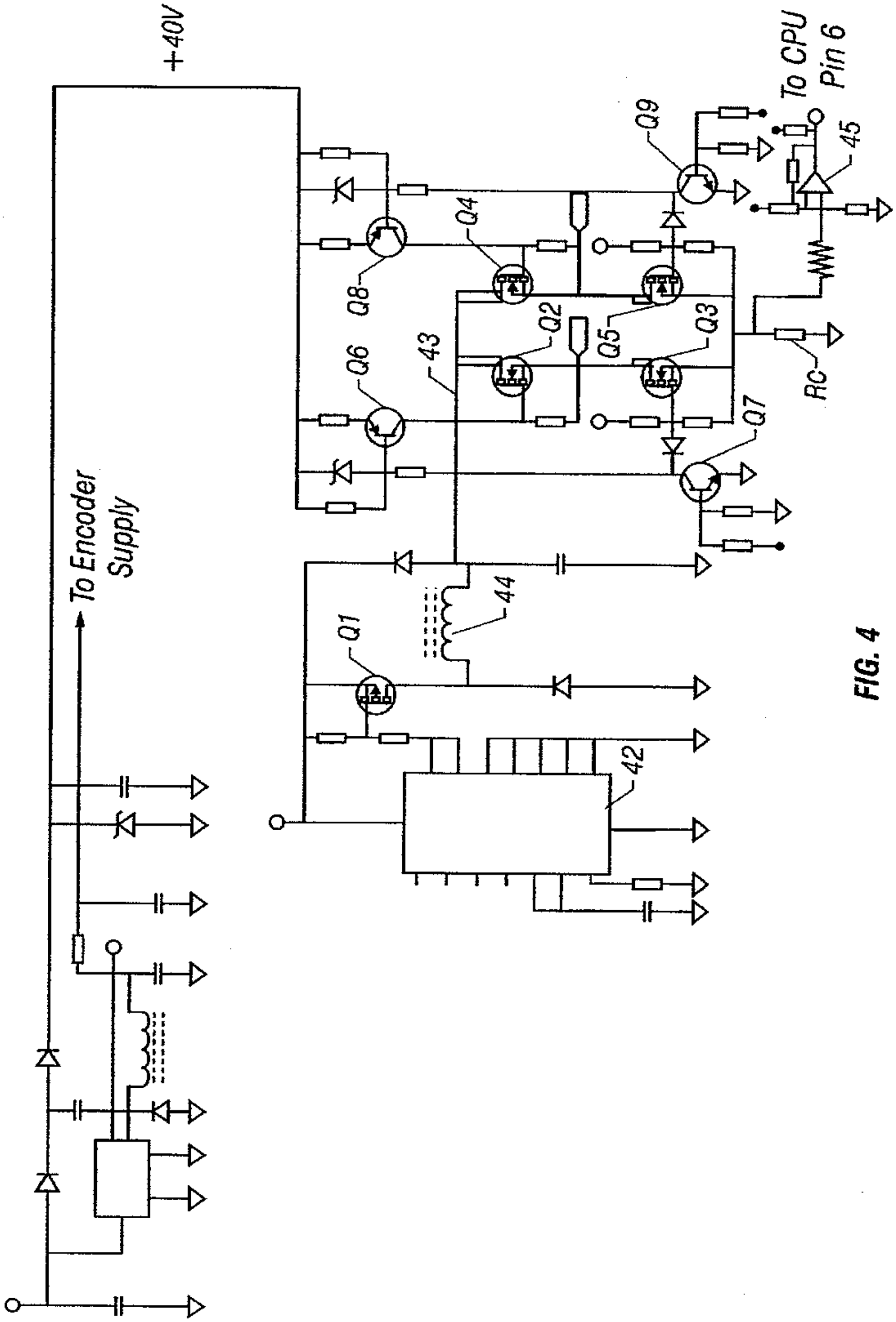

+40V
To CPU
Pin 6
To Encoder
Supply
Q1
Q2
Q3
Q4
Q5
Q6
Q7
Q8
Q9
42
43
44
45
Rc

FIG. 4

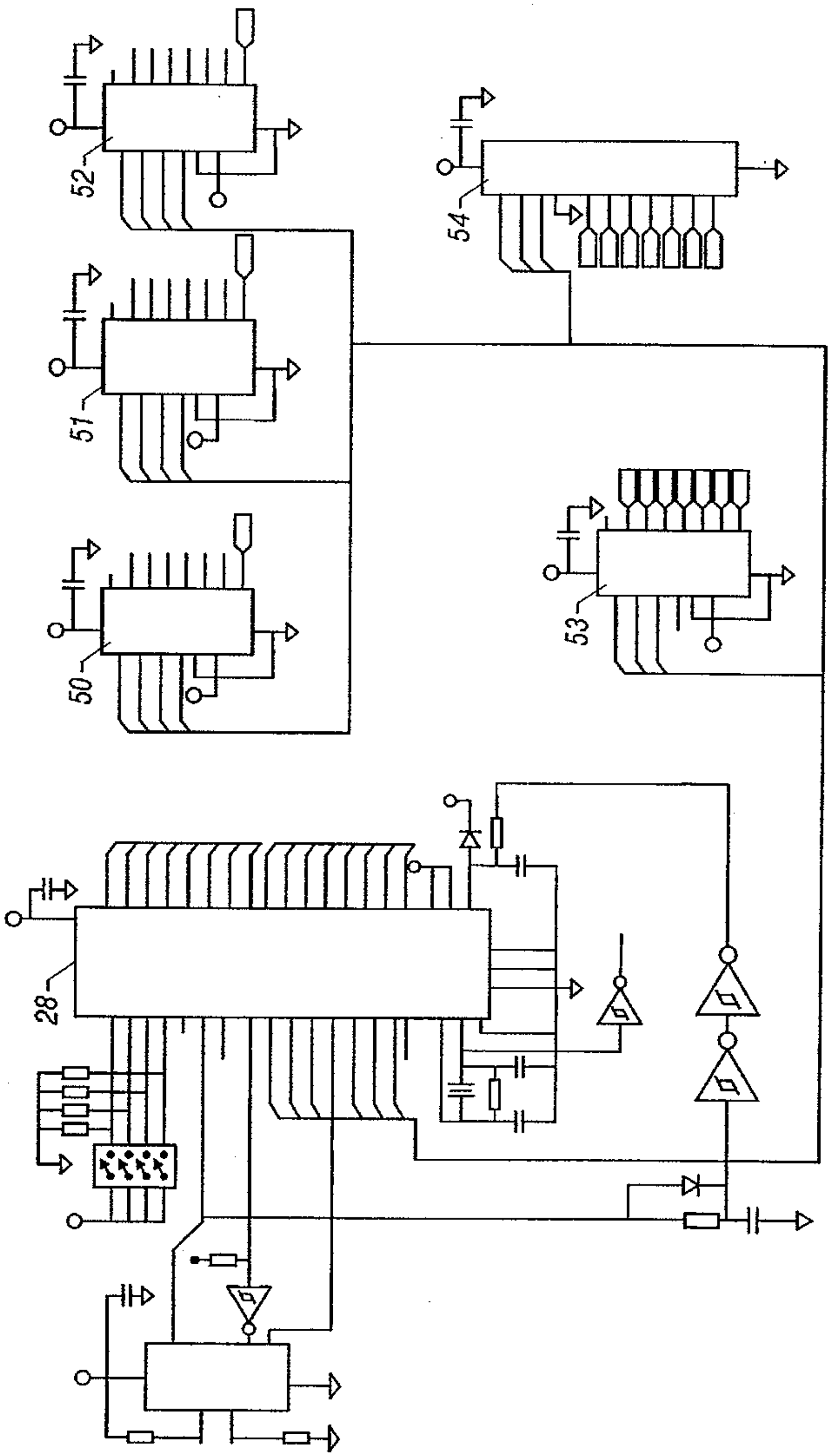


FIG. 5

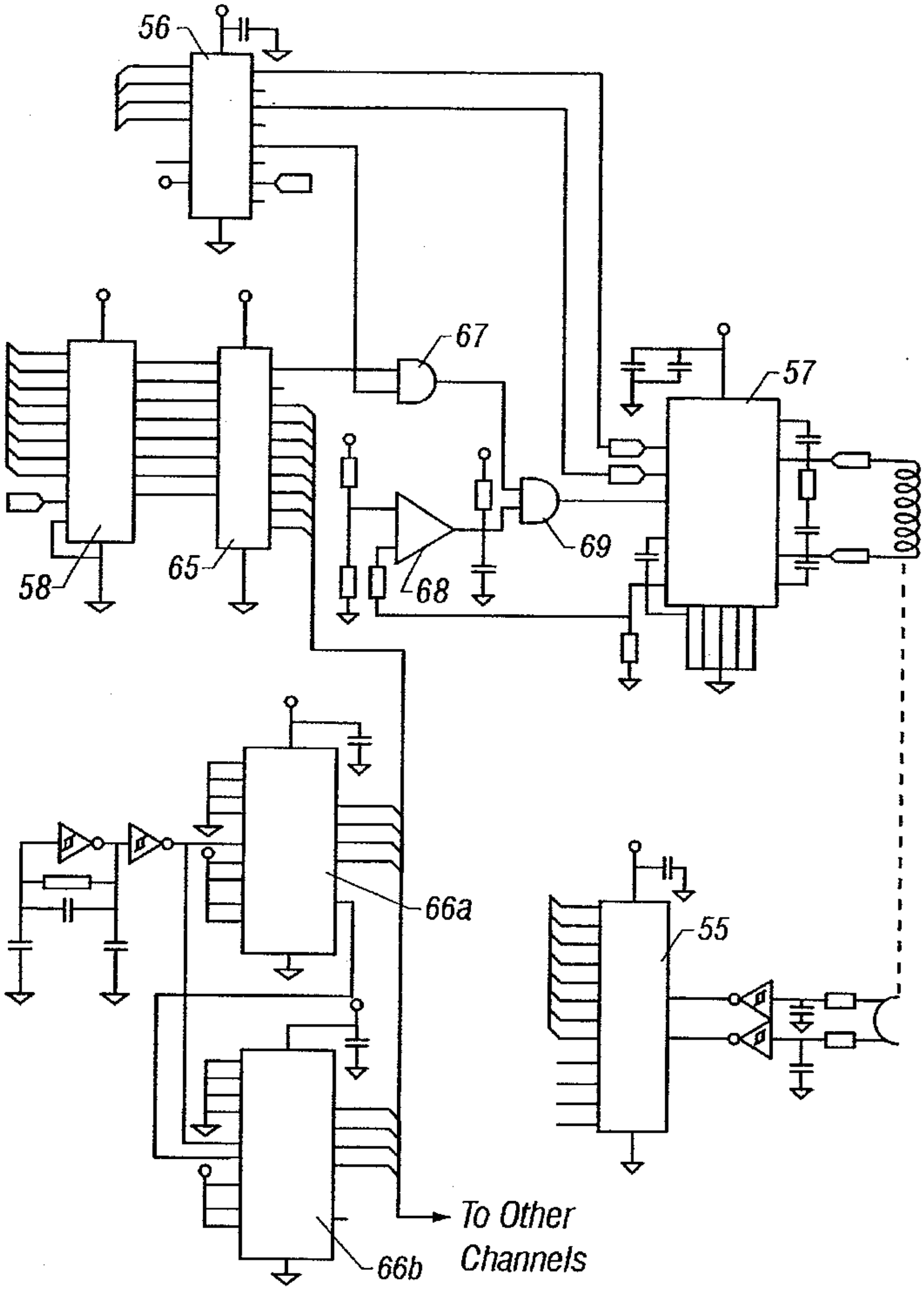


FIG. 6

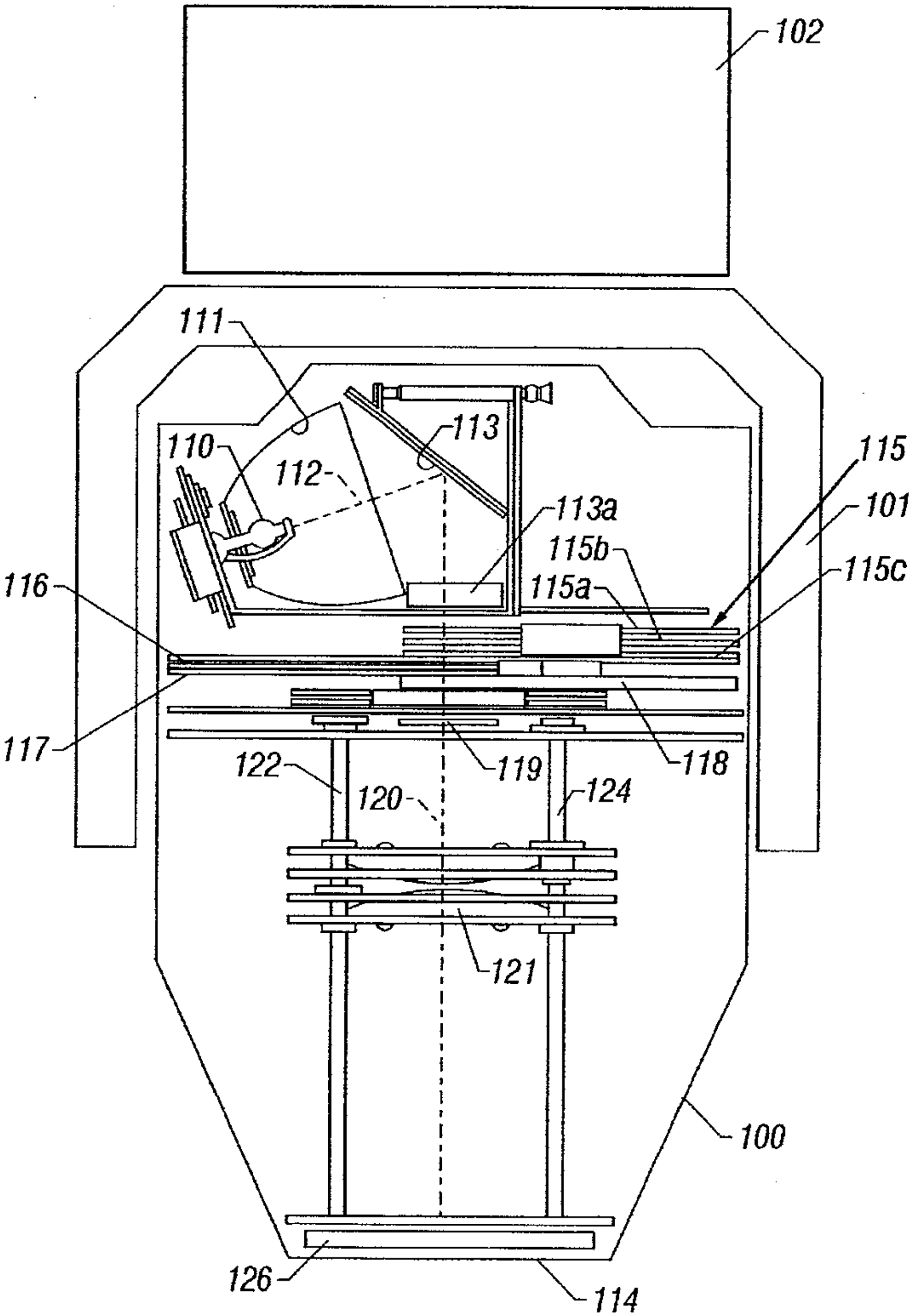

102
111
110
112
113
113a
115b
115a
115
101
115c
116
117
122
120
119
124
118
121
100
126
114

FIG. 8

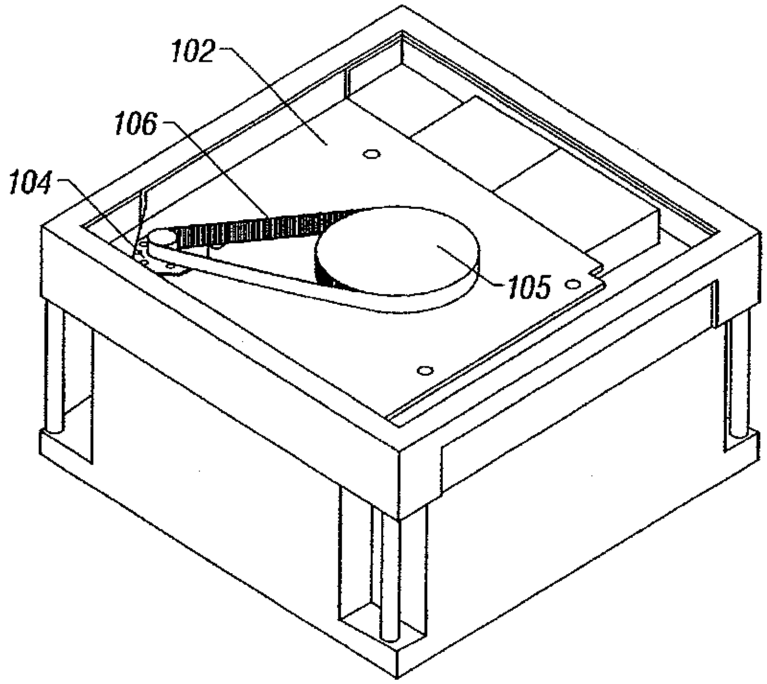


FIG. 9

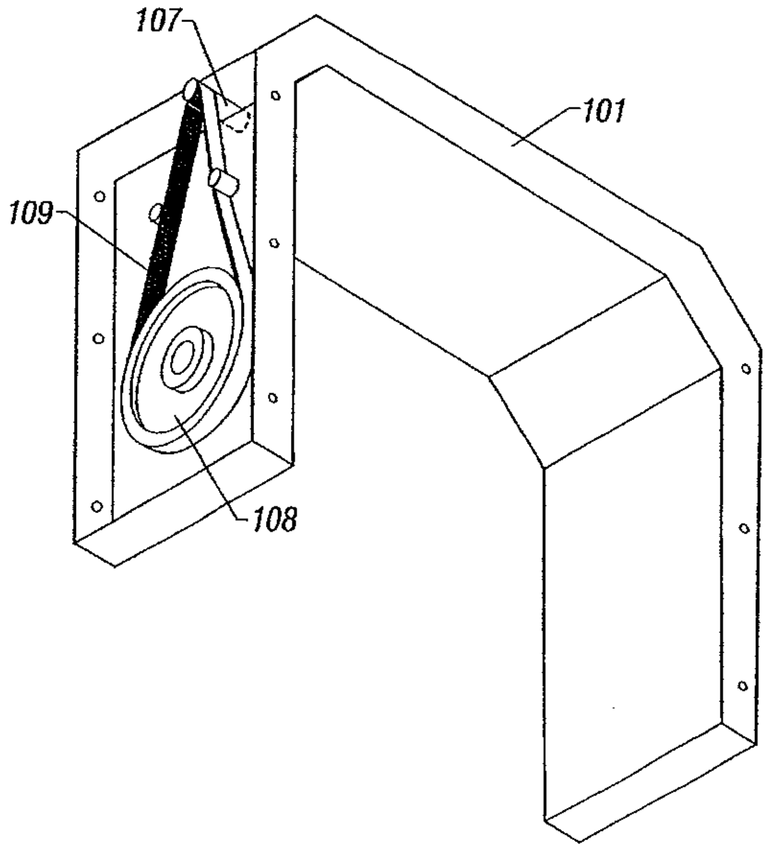


FIG. 10

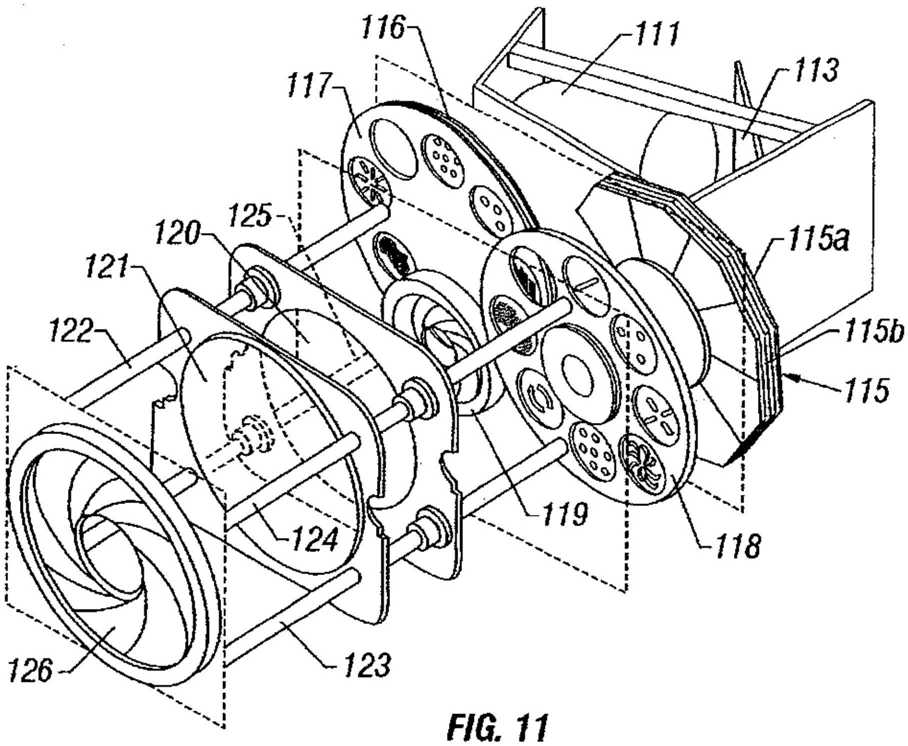

116
111
113
117
125
120
121
115a
122
115b
115
119
124
118
126
123

FIG. 11

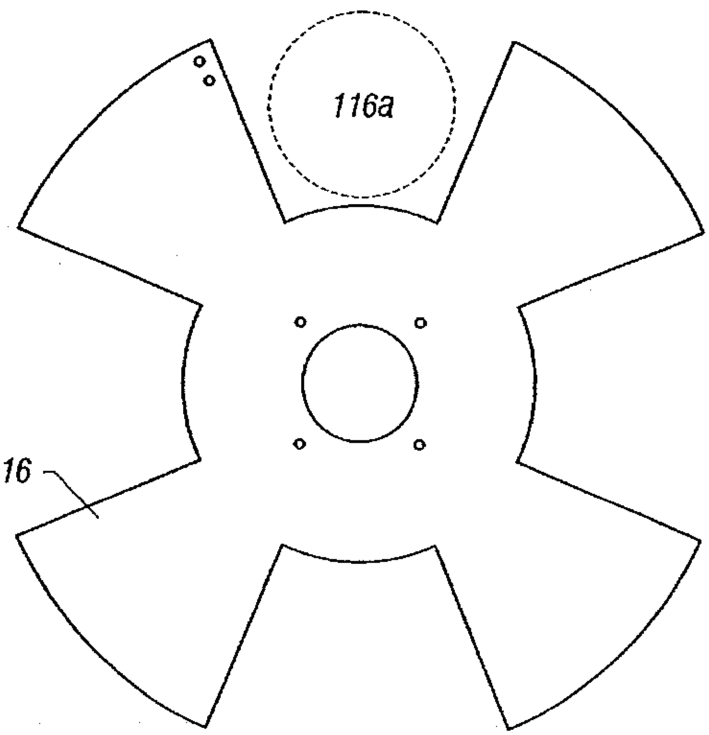

116a
116

FIG. 13

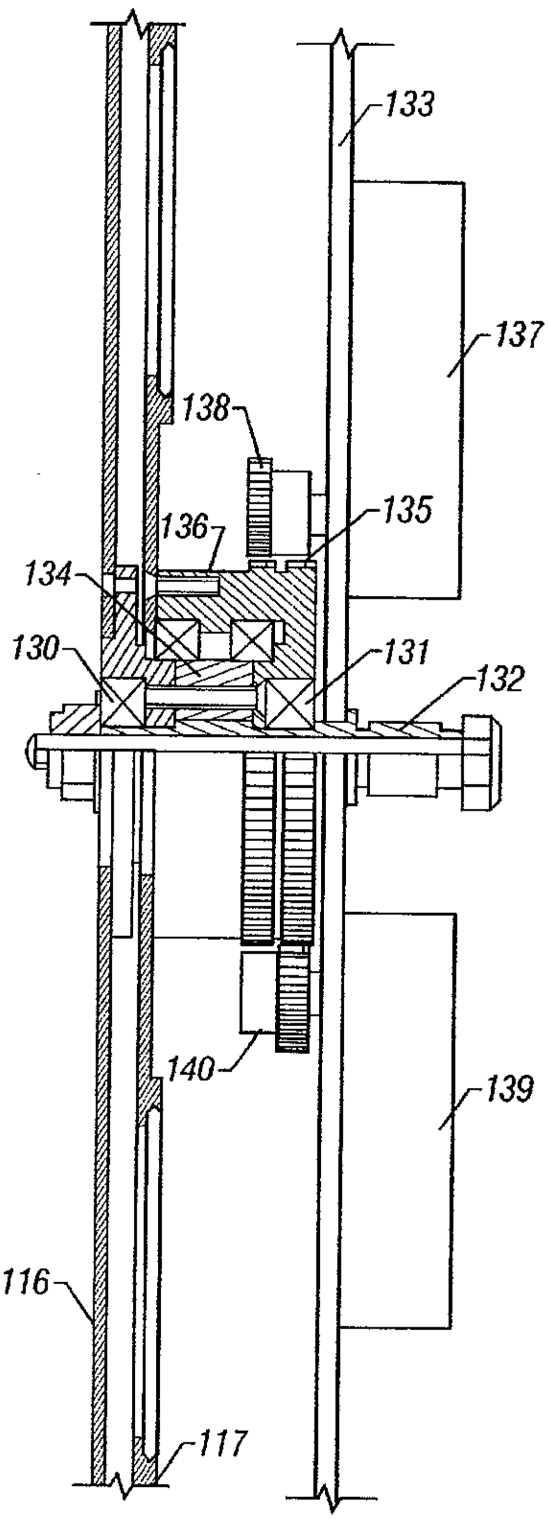


FIG. 12

STAGE LIGHTING LAMP UNIT AND STAGE LIGHTING SYSTEM INCLUDING SUCH UNIT

This is a continuation of application Ser. No. 09/313,418 filed May 17, 1999, now U.S. Pat. No. 6,326,741, which is a continuation of application Ser. No. 08/994,036, filed Dec. 18, 1997, now U.S. Pat. No. 5,921,659, which is a divisional of application Ser. No. 08/576,211, filed Dec. 21, 1995, now U.S. Pat. No. 5,788,365, which is a continuation of application Ser. No. 08/077,877, filed Jun. 18, 1993, now U.S. Pat. No. 5,502,627.

This invention relates to stage lighting and is particularly concerned with the control of multiple functions of a lamp.

It has already been proposed to incorporate in a lamp unit a plurality of different functions, such as colour changers, focusing lenses, iris diaphragms, gobo selectors and pan and tilt mechanisms which are controlled from a remote console. Stage lighting systems have as a result reached very high levels of complexity requiring a very complicated main control console and lamp unit constructions. The use of microprocessors, both in the console and the lamps has become conventional as increasing complexity makes it more difficult to produce and subsequently maintain a system which uses hard wired logic or analog controls. In such systems the microprocessor in the console is used to allow the user to set up lighting cues and to control the sending of appropriate data to the lamp microprocessors. The lamp microprocessors are also involved in controlling communication between the console and the lamps, and also have to control a plurality of servomotors which drive the various functions of the lamps.

It is one object of the present invention to provide a lamp microprocessor and servo-control arrangement which allows complex functions to be carried out.

It is another object of the invention to provide a lamp control system in which control of pan and tilt movements of each lamp can be carried out in rapid and efficient manner, enabling large groups of lamps to make co-ordinated movements.

It is yet another object of the invention to provide each lamp in a stage lighting system with a means for quickly interrupting its light beam and quickly re-establishing the beam so that a group of lamps can be made, when required to flash in synchronism.

In accordance with one aspect of the invention there is provided a lamp unit for connection to a remote control console for the control of a plurality of different functions of the lamp, said unit comprising a main processor circuit, associated with a communication controller for accepting message data from the console, a plurality of servo-controls for operating said functions of the lamp, and a plurality of co-processors which are connected to the main processor circuit so as to be supplied thereby with desired value data for the various lamp functions, said servo-controls being controlled by said co-processors.

In the case of pan and tilt controls where close control is required throughout the movement of the lamp from an initial position to a new position, one of the co-processors is assigned solely to the control of movement about each axis. Other functions can share a co-processor.

The main processor circuit of the lamp is preferably programmed to accept data from the control console defining not only a target position for any function, but also a duration over which the function is to be executed. In this case the main processor circuit divides the "journey" into segments and updates the target position data passed to the associated co-processor at intervals.

In accordance with another aspect of the invention, there is provided a lighting control apparatus comprising the combination of a main control console for accepting user input relating to required beam movements, a plurality of independently operable lamp units situated remotely from the console, each of the lamp units incorporating a servo-mechanism for automatically moving the lamp beam about two mutually transverse axes to a desired angular position and data communication means connecting the console to the lamp units for the transmission of desired position data to the lamp units, the desired position data being transmitted in the form of a set of three dimensional linear co-ordinates defining a point in space through which the lamp beam is required to pass, and each lamp unit including a calculating device for calculating the desired angular position from the desired position data and supplying the servo-mechanism with such desired angular position.

In addition to the "point at" mode of operation mentioned above, additional modes may be specified in which the lamps point away from the specified point or in which they all point in the same direction parallel to a line between a fixed position in the co-ordinate system and the specified point.

Conveniently, all the data concerning the positions and orientations of the individual lamp units within the co-ordinate system is stored in a set-up file kept on a hard disk drive in the console. When the same lighting set-up is used at different venues, where it is impossible to set the frame which carries all the lamp units at exactly the same position as that for which the set-up was designed, offset data can be input at the console and either used within the console microcomputer to correct the position data stored during set-up as it is sent out, or such data can be sent to all or the lamp units over the network and stored there, to enable the corrections to be made in the individual lamp processor units.

In accordance with another aspect of the invention, a stage lighting unit comprises a housing, a light source within said housing, an optical system for forming light from said light source into a beam, a rotary shutter device having a plurality of blades, said shutter device being rotatably mounted in the housing so as to cause said blades to pass through and obstruct said beam as the shutter device rotates, a motor for rotating said shutter device and a servo-control for controlling said motor in accordance with data received in use from a remote control console.

The invention also resides in a stage lighting system incorporating a plurality of lighting units as defined above controlled by a common remote control console via data communication means, whereby the rotary shutter devices of all the units can operate in synchronism.

An example of the invention will now be described with reference to the accompanying drawings, in which:

FIG. **1** is a block diagram of a stage lighting system;

FIG. **2** is a block diagram of the internal circuitry of one of a plurality of lamp units in the system of FIG. **1**;

FIGS. **3** and **4** are more detailed circuit diagrams showing a pan motor drive control forming part of the internal circuitry of the lamp;

FIGS. **4** to **7** are detailed circuit diagrams showing a rotary shutter motor drive control forming part of the internal circuitry of the lamp;

FIG. **8** is a diagrammatic, part-sectional view of one of the lamps;

FIG. **9** is a perspective view of a pan movement drive arrangement;

FIG. **10** is a perspective view of a tilt movement drive arrangement;

FIG. 11 is a diagrammatic perspective view of the internal moving parts or the lamp;

FIG. 12 is a sectional view showing the drive arrangement for a shutter and a gobo wheel forming part of the lamp; and

FIG. 13 is an elevation of a shutter wheel forming part of the lamp.

Referring firstly to FIG. 1, the system consists basically of a console unit 10, a signal distribution unit 11 and a plurality of lamps L1, L2, L3 . . . , L31, L32, L33 . . . , L61, L62 . . . individually connected by twisted pair data communication links to the distribution unit.

The console unit 10 has an array of switches, slider potentiometers, rotary digital encoders and other user actuable input devices (not shown) and a display indicated at 13. These are all connected to main console cpu 14 (an MC68020 micro-processor) which has the task of receiving inputs from the user actuable input devices and controlling the display. Both tasks are assisted by separate co-processors which directly interface with different parts of the console.

The main cpu can communicate with a hard disk drive unit 15 via a SCSI bus 16 which also connects it to the distribution unit and to an external SCSI port 17, through the intermediary of which the console can, if required be connected to a personal computer. The user controls can be used in setting up a sequence of cues in advance of a performance, the sequence being stored in a cue file on the hard disk drive unit 15. The sequence can be recalled during the performance to enable the various stored cues to be executed. Direct manual control of the lamps from the console is also possible as is manual editing of cues called up from the hard disk. The main console cpu 14 creates messages to be sent to the individual lamps, each message comprising a fixed number of bytes for each lamp. The messages contain data relating to the required lamp orientation, beam coloration, iris diaphragm diameter, gobo selection and rotation, zoom projection lens control and opening or closing of a shutter included in the lamp. A block of the RAM of the main cpu is set aside for the storage of these messages, the block being large enough to contain messages for 240 lamps, being the largest number which can be controlled via the distribution unit. Where it is required to control more than ~240 lamps additional distribution units can be connected to the SCSI bus and extra main cpu RAM reserved for message storage. When any message data is changed the main cpu 14 sets a flag in the RAM block which is detected at a given point in the main cpu program loop and interpreted as a signal that the changed message data is to be transferred to the distribution unit 11.

The distribution unit 11 has a main cpu 19 which controls reception of data from the SCSI bus interface and distribution of such data to up to eight blocks of dual,port memory DP1, DP2, DP3 . . . via an eight bit data bus 20. The cpu 19 is alerted to the waiting message data when cpu 14 selects the distribution unit. The cpu 19 then supervises byte by byte transfer of the message data which it routes to the various blocks of dual port memory.

For actually sending out the message data to the lamps, there are a plurality of serial communication controllers SCC1 to SCC30, SCC31 to SCC60 etc, there being thirty serial communication controllers associated with each block of dual port memory. A further cpu DCPU1, DCPU2, etc is associated with each block of dual port memory and distributes message data transferred to the dual port memory to the individual serial communication controllers and the messages are transferred to the lamps. Each serial communication controller in the distribution unit includes a line driver which can be disabled except when data is to be transmitted. Enabling of the driver can cause a spurious signal to be transmitted over the data link. To allow such spurious signals to be identified and ignored, a two-byte gap is left between enabling the line driver and commencing transmission of the message data for the channel in question.

This will be described in more detail herein. All asynchronous serial communication systems require framing information to synchronize the reception process. This has been typically done in the prior art using start bits and stop bits.

The present invention preferably uses FMO coding in which the data is transmitted as one cycle of the carrier frequency for a zero or as a half cycle of the carrier frequency for a one. When the line has been idle, no waveform at all is present. When the line drivers are first enabled, an arbitrarily short pulse will usually appear on the line, due to lack of synchronization between the data signal and the enabling signal. This short data pulse could be misinterpreted as a start bit, for example and if so it would disturb later framing.

The present invention avoids any problems from this arbitrarily short pulse. To avoid this, the present invention uses a timer on the receive line, set to the time needed to receive two bytes on the serial data line. This timer is restarted whenever a byte on the data line is detected.

Each time the timer interrupt occurs, the number of bytes received is checked against the number of bytes in a valid data frame. If the number is incorrect, then the count is cleared and the message is discarded. If correct, the information is passed to the main program loop by setting a flag variable.

When the data line is first enabled, the distribution box has an internal delay of at least two byte times, which must elapse before any data will be sent. Any data received by the lamp will therefore be discarded as noise by the timer interrupt routine. After that, the real data can be safely sent down the line since the start bit of the first byte will be received correctly. When the transmission is completed, the line drivers will be disabled again.

Each of the cpus eg DCPU1, transfers data from the associated dual port RAM DP1 to the serial communication controller SCC1 to SCC30 with which it is associated one byte at a time, ie the first byte for SCC1 is transferred followed by the first byte for SCC2 and so on, each serial communication controller commencing transmission as soon as it has received its byte of data. The serial communication controllers operate to transmit data at 230.4 Kbps so that it takes about 35 μs to transmit each byte. Transfer of data from the dual port RAM DP1 to the serial communication controllers is, however, at a rate of several Mbps, so that the transmissions from all the serial communication controllers are almost simultaneous. The cpu DCPU1 is not required to monitor the transmission of data by the serial communication controller, but utilizes a software timer to commence transfer of the second byte to the serial communication controllers. This timer is started when transfer of the byte of data to the last serial communicationcontroller SCC30 has been completed and its timeout duration is slightly longer than the byte transmission time, say 40 μs. Transmission of all the messages takes about 1.5 ms out of a distribution unit main program loop duration of 4 ms.

As shown in FIG. 2, each lamp includes a serial communication controller 20 which controls reception of message data from the individual data link connecting it to the distribution unit 11. The receipt of any signal from the data link causes an interrupt of the lamp main cpu 21 (another

MC68000) and the cpu **21** then controls acceptance of the signals. A timer 22 times the gaps between bytes received from the data link and this timer causes another interrupt on time-out. The time-out time of the timer is between the times taken to transmit 1 and 2 bytes, so that time out always occurs following a spurious signal caused by line driver enabling. The time-out interrupt causes the cpu **21** to inspect the total number of bytes received since the initial interrupt and if this is less than the expected number of bytes (which is constant) the message is ignored. The time-out interrupt also resets a software data pointer to the beginning of a receive buffer in readiness for the next transmission.

The cpu **21** operates in accordance with programs stored in the lamp cpu ROM. On receipt of a message of valid length, a program variable representing the number of messages received since the lamp program was last started is incremented and the main program loop of the lamp cpu checks this variable every 16 mS. If the variable has changed since the last check, the data in the receive buffer is compared with corresponding values of variables representing current "desired values" of the various lamp function parameters. For example the receive buffer may contain two bytes representing the x, y and z co-ordinates of a point in an orthogonal three dimensional frame of reference, through which point it is required that the axis of the lamp beam should be directed. If the values of the corresponding byte pairs in the receive buffer and the desired value variables already contained in the cpu RAM are the same, no action is taken in respect of the control of the motors which control pan and tilt action of the lamp (to be described in more detail hereinafter).

As shown in FIG. **2**, the main lamp cpu **21** communicates via serial data links **25***a*, **25***b*, **25***c* and **25***d* with four servo-control co-processors **26**, **27**, **28** and **29**. Each of these co-processors is a TMS77C82 cpu. Co-processors **26** and **27** respectively control pan and tilt operation, and each of the co-processors **28** and **29** can control up to six different dc servo-motors operating different functions of the lamp.

Before proceeding with a more detailed description of the circuitry and operation of the lamp electronics, some detail will be given of the various functions of the lamp. FIG. **8** shows the relative positions of a plurality of independently operable beam characteristic control elements within the lamp housing **100**. The lamp housing is pivotally mounted on a U-bracket **101**, which is itself pivotally mounted on a mounting base **102**. FIG. **9** shows the mounting base **102** which incorporates a pan drive motor/gearbox/optical encoder arrangement **104** which drives a gear **105** attached to the U-bracket via a reduction toothed belt drive **106**. FIG. **10** shows how, within the hollow structure of the U-bracket **101**, there is mounted a tilt drive motor/gearbox/optical encoder **107** which drives a gear **108** attached to the lamp housing via another reduction toothed belt drive **109**.

As shown in FIGS. **8** and **11**, within the lamp housing, a light source **110** is mounted within an ellipsoidal reflector **111** providing a light beam with an axis **112** which is reflected by a mirror **113**, which is a dichroic mirror that reflects only visible light and passes ultra voilet and intra red light, the reflected light passing out through an opening **114** at the opposite end of the housing. The reflector **111** has a generally cup-shape surrounding the bulb **110**. According to one aspect of the invention, the axis **112** has an angle pointing in a direction rearward relative to a perpendicular to the central axis **120** of the lamp unit. If the reflector is located as shown, such that an outside edge of the reflector is generally parallel to a rear end of the housing, the optimal packing efficiency is achieved. As shown in FIG. **8**, this

allows the reflector to be most efficiently packed into the available space. The reflected beam from the mirror **113** passes firstly through a collimating lens **113***a*, and then the colour changer **115** which comprises dichroic filters having differing transmission characteristics mounted on co-centered three filter disks **115***a*, **115***b* and **115***c* rotable around a common axis of rotation. Each disk has nine different filters on it and one blank space around its periphery, so that up to 1000 different combinations of filters can be positioned across the beam by selective positioning of the three disks (although not all of these combinations are necessarily useful as some may block all visible light). The blank space of each of the disks can be used to eliminate any color changing characteristic of that disk. These disks are driven by three of the dc servo-motors. Next the light beam passes through the plane of a bladed shutter **116** (shown in FIG. **13**) and a first gobo wheel **117** which has various gobos mounted in or over circular holes therein. As shown in FIG. **12** described in more detail hereinafter, two motors are committed to driving the shutter **116** and the gobo wheel **117** respectively. Next, there is a second gobo wheel **118** on which there are mounted a plurality of gobos which are rotatable relative to the wheel **118**. There is one motor (not shown) for driving the gobo wheel **118** and another for rotating the gobos mounted thereon through a gear arrangement (not shown). Next along the light beam is a beam size controlling iris diaphragm **119** driven by another motor (not shown). Two further motors (not shown) drive two lens elements **120**, **121** along guides **122**, **123** parallel to the beam axis using lead screws **124**, **125**. The lens elements form a simple two element zoom lens controlling the spread and focus of the beam. Finally, an outer iris diaphragm **126** is provided adjacent the opening **114** and this is driven by a further motor (not shown). In the example described, therefore only eleven channels are actually employed.

Referring now to FIG. **12**, the shutter **116** is rotatably mounted on bearings **130**, **131** on a shaft **132** fixed to a mounting panel **133** which is secured to the housing. The gobo wheel **117** is rotatably mounted on bearings on a tubular shaft **134** which acts to space the shutter **116** from a first drive gear **135**. The gobo wheel **117** is actually mounted on a second drive gear **136**. The shutter motor **137** (which is combined with a reduction gearbox and an optical encoder) is mounted on the panel **133** and drives a pinion **138** meshed with the first gear **136**. Similarly motor **139** drives a pinion **140** meshed with the second gear **136**. The shutter has four blades arranged symmetrically around its axis, with the blades and the gaps between them each subtending 45 degrees at the axis. The blades and the gaps between them are wide enough to block or clear the entire cross-section of the beam, shown in FIG. **13** at **116***a*.

Turning now to FIGS. **3** and **4**, the co-processor **26** is shown providing an eight bit data output to a d/a converter **40** (FIG. **3**) the output of which is amplified by an operational amplifier **41** and supplied to the "COMPEN" terminal of an LM3524 pulse width modulator ic **42** (FIG. **4**). The ic **42** control a P-channel enhancement mode MOSFET Q1 which, when switched on, connects a 24V supply to a motor supply bus **43** through the intermediary of an inductor **44**. The motor is connected in a bridge formed by two push-pull pairs of MOSFETs Q2, Q3 and Q4, Q5. These four MOSFETs are driven by respective driver transistors Q6, Q7, Q8 and Q9. Transistors Q7 and Q9 are respectively controlled by "LEFT" and "RIGHT" outputs taken from the co-processor **26**, so that FETs Q2 and Q5 or FETs Q3 and Q4 are biassed to conduct. Transistors Q6 and Q8 are driven from a 40V supply rail so as to ensure that FETs Q2 and Q4

are turned hard on when conductive, thereby ensuring minimum power dissipation in these devices.

The two FETs Q3 and Q4 are connected to the return bus via a current sensing resister RC, which supplies a current related signal to a voltage comparator **45** with hysteresis to provide an input to the A6 input terminal of the co-processor **26** when the current exceeds a predetermined limit. This enables the co-processor to reduce the power applied to the motor to maintain it within safe operating limits.

The optical encoder of the pan motor provides two digital outputs in quadrature, these outputs being cleaned up by interface circuits and applied to two inputs of an HCTL-2016 counter ic **46** intended specifically for use with quadrature type encoders. The counter **46** counts up when the pulses are in one relative phase relationship and down when the opposite phase relationship exists. It therefore maintains a count-state related to the motor shaft position and hence the pan angle of the lamp. This count-state is applied to the C0 to C7 terminals of the co-processor **26**. The co-processor **26** also receives "desired value" data from the main lamp cpu **21**, via a 75176 ic **47** (which in fact serves both co-processors **26** and **27**). The ic **47** is used to control the transmission of data between the main lamp cpu and the co-processors. Normally the ic **47** is set to receive data from the cpu **21** and pass it to the two coprocessors **26** and **27**. At power-up or when the main lamp cpu **21** transmits a "break" command, the co-processor **26** is reset by a circuit **48**. The coprocessor **26** has a cycle time of 1 mS and on receipt of new data it determines the distance to be travelled and then increases the "desired position" value which is compared with the actual position count by one sixteenth of the required change on each successive iteration of its control loop.

The desired value signals passed from the cpu **21** to the co-processor **26** are also time-sliced, being incremented every 16 mS. When new position data is transmitted to the lamp it is accompanied by data representing the length of time over which the movement is to be spread. The data is received, as mentioned above, in the form of two byte numbers respectively representing the x, y and z co-ordinates of a point in a Cartesian co-ordinate system. During initial setting up of the system, each lamp is sent data which informs its cpu **21** of its position in the coordinate system and also of its orientation.

On receipt of a new set of "point at" co-ordinates, the cpu **21** undertakes a "time-slicing" operation to determine how data should be passed to the coprocessors **26** and **27**. First of all, it determines how many 16 mS loops will take place in the time duration determined by the data contained in the massage received by the lamp and sets up a variable U equal to the reciprocal of this number. A travel variable P is initialised to zero and the total distance to be travelled is determined for each of the pan and tilt movements. Thereafter, on every iteration of the 16 mS loop the travel variable P is incremented by the reciprocal variable U, the result is multiplied by the total travel required and this is added to (or subtracted from) the previous desired value before transmission to the coprocessor **26** or **27**. When the variable P exceeds unity, the target has been reached.

The message sent to the lamp may include a flag indicating whether travel is to occur in a linear fashion as described above or have a sinusoidal profile imposed on it. In the latter case the value of P is modified as follows:

$$P' \sin(2^*P)+0.5^*(P>0.5) \text{ the latter term being 0 or 1}$$

The main cpu **26** must next convert the x,y,z values into pan and tilt value data for passing to the co-processors **26** and **27**. The cpu first carries out a linear transformation of the absolute x,y,z co-ordinates into co-ordinates x',y',z' relative to the lamp's own frame of reference using the data supplied during initial set up. The ratio of the transformed x' and y' values is calculated as a 16-bit integer, which is used as an index to an ARCTAN table stored in ROM to obtain a value for the desired pan angle. To find the tilt angle, it is first necessary to establish the radial position of the target point in the transformed horizontal plane by calculating the square root of the sum of the squares of the co-ordinates x' and y'. In carrying out this calculation it is necessary to detect an overflow condition which exists if the sum of the squares is a 33 bit number. If this condition is detected, each square is divided by four and a new sum is formed, an overflow flag being set to indicate that overflow has occurred. The square root is found by up to sixteen steps of successive approximation and the result is doubled if the overflow flag was set during the calculation. The resulting square root is divided by the value z' and the result is applied as before to the ARCTAN table to determine the tilt angle. The results obtained represent the new pan and tilt positions to which the lamp is to be moved.

The arrangement described for sending out x, y and z co-ordinate data instead of pan and tilt angle data is highly advantageous in that it enables the console main cpu load to be significantly reduced and also makes it very easy for a console operator to control light beam movements. It is frequently required for a group of lamps to be used together to illuminate a single performer. Where the performer moves from one position on stage to another it is required for all the lamps to change position simultaneously to follow. If the system involved transmission of pan and tilt angle data, this data would be different for every lamp in the group. It would have to be set up by the console operator and stored in cue files on the hard disk drive unit **15**. This would be a very time consuming operation as the pan and tilt angles for each lamp would have to be established and recorded individually. The cue record would need to be of considerable size to record all the different data for each lamp. With the arrangement described above, however, only the x,y,z co-ordinate data needs to be stored and when the cue is recalled the same data is sent to each of the lamps in the group.

Whilst it is theoretically possible to use stored cue data in x,y,z coordinate form and to use the console main cpu **14** to calculate the pan and tilt angles to send to the lamps, this would be unsatisfactory as the calculations involved would impose a very heavy load on the cpu **14**, particularly where a large number of lamps in several different groups had to be moved as the result of a single cue.

As described above a "point-at" mode is envisaged as the normal operating mode. However, other modes of operation are also envisaged. For example, the lamp could be instructed to point away from the point specified or to point in a direction parallel to a line joining a fixed point (eg the origin of the co-ordinate system) to the point specified. These "point-away" and "point parallel" modes would be selected by means of flags included in the data transmitted to the lamps.

The arrangement described enables the lamps to be very precisely synchronised. The data is transmitted from the distribution unit to all of the lamps simultaneously and each lamp can start to respond at the end of the message. This enables very precise direction of all the lamps to a moving point in "point-at" mode and very clean parallel sweeps to be made in "point parallel" mode.

It should be noted that the use of x,y,z co-ordinates is also very advantageous in situations where a prearranged lighting

performance is to be used in several different venues. The pre-loaded gantries or trusses used for such touring performances cannot always be mounted at exactly the required positions relative to the stage because of local conditions. In this case all that is needed is for offsets data to be sent to the lamps at set-up time to enable each lamp cpu to correct its position data. No editing of the individual pre-recorded cues is necessary as it would be in the same circumstances if pan and tilt data were stored.

As part of the set-up procedure for each performance it is necessary to initialise the values of the actual pan and tilt angle count-states, since encoders of the type used do not give any absolute position data. This is accomplished by driving the lamp to an end stop in one direction for each movement. The lamp is driven back to a predetermined number of counts and the counters are reset to zero at this position.

Turning now to FIGS. 5 to 7, the circuitry for controlling the individual dc servo-motors inside the lamp is more complex as each co-processor has to deal with up to six servo-motors. As shown in FIG. 5, the coprocessor **28** controls a number of data routers **50** to **54** which determine which channel is being controlled at any given time. The router **50** co-operates with six HCTL-2016 counters **55** which count the quadrature pulse outputs of the respective encoders, to determine which of the counters should supply its count-state to the co-processor **28**. Router **51** controls individual resetting of the counters **55**. Router **52** co-operates with a 74HC175 ic **56** (one for each channel) to determine which L6202 ic motor controller **57** is enabled and also routes "RIGHT" and "LEFT" signals from the co-processor to the circuits **57**. Router **53** controls routing of position error data calculated by the co-processor **28** for each channel to latches **58** (one for each channel) at the input of pulse width modulator circuits for controlling the motor controllers **57**. This error data is actually passed to the latch **58** in an inverted form, so that the larger the error, the smaller the value passed is. Router **54** routes various digital sensor signals to a sensor input of the co-processor, Such sensors are utilized by some of the channels to indicate when the moving part in question is in a datum position. This is required for the gobo wheels, the colour wheels and the shutter, but not for the iris diaphragms or lenses which can be moved to end stop positions. During datum set-up the sensors (optical sensors sensing a hole or flag or Hall effect sensors) are detected and the HCTL counters are reset.

As co-processor **28** has only 256 bytes of internal memory, extra memory is required for each channel to store program variables. The RAM selection control circuit is shown in FIG. 7. The memory ic **59** (an HM6116LP ic) has 11 address lines of which eight are connected to the co-processor write bus via a latch circuit **60** and the remaining three of which are connected to spoare outputs of three of the ics **56**. Spare outputs of the selectors **50, 51, 52** are connected to control terminals of the memory ic and a spare output of the selector **53** is connected to an output enable terminal of the latch circuit **59**. Thus a particular address in the memory ic can be selected by the co-processor by first setting the ics **56** and the selectors **50, 51, 52** to appropriate states and then outputting the lower bytes of the address to latch **60** whilst output from latch **60** is enabled. Two further eight bit latches **61** and **62** provide temporary storage for data to be written to and data just read from the memory ic **59**. When neither reads nor writes are required the memory data bus is tri-stated. Bus contention is thus avoided.

Circuit **57** actually controls the motor current, but it in turn is controlled by a pulse width modulator circuit, com-

prising the latch **58** and a digital comparator **65** which compares the contents of latch **58** with the count-state of an 8-bit continuously running counter **66***a*, **66***b* serving all channels. The comparator output goes high when the count-state exceeds the latch contents, so that if the latch content is low the comparator output is high for a high proportion of each cycle of the counter .**66***a*, **66***b*. The output of the comparator **65** is ANDed with an enable output from ic **56** by a gate **67** and then with the output of an overcurrent detector circuit **68** by another gate **69**.

When a new target value for one of the parameters controlled by coprocessor **58** arrives in the receive buffer, and it is associated with execution duration data (this may apply to lens movements, colour changer movements, gobo movements and iris diaphragm movements, but not shutter movements) the cpu **21** handles time slicing as in the pan and tilt operations. Since several channels are controlled by each coprocessor, however, no interpolation by the co-processor is used. Instead each channel has its error checked and a new value written (if necessary) to latch **58** every 12 mS

In the case of the shutter, the message received by the lamp merely includes a shutter open or shutter closed command. When the required shutter status changes, the main cpu merely increases the target shutter angle by 45 degrees (in the case of a four bladed shutter) and passes the new value to the co-processor.

This arrangement enables the shutters of some or all of the lamps to be operated in synchronism. Moreover, the console cpu **14**, can operate to update the shutter open/closed instructions at regular intervals to obtain a stroboscopic effect, synchronised for all the lights.

What is claimed is:

1. A lighting system, comprising:

a main console unit including a plurality of user controls respectively representing controls for a plurality of lighting units, a main processor, and an output port formed to connet to a bus that carries controls for each of said plurality of lighting units;

a distribution unit, adapted for connection to said bus to receive said controls for each of said plurality of units, and including a plurality of connection parts for said plurality of lighting units, said distribution unit including a processing part that receives said information from said bus and distributes said information to said plurality of connection parts.

2. A system as in claim **1**, wherein said processing part within said distribution unit includes dual port memory which receives information for specified lighting units at one port thereof, and provides an output at another port thereof.

3. A system as in claim **2**, wherein said bus is a SCSI bus.

4. A system as in claim **2**, further comprising, within said processing part, a plurality of said dual port memories, and a plurality of processors, wherein each of said dual port memories is associated with one of said processors.

5. A system as in claim **4**, further comprising a plurality of serial communication controllers, receiving outputs from said processors, and sending communication to the lamps.

6. A system as in claim **5**, wherein said serial communication controllers also receive information from said lamps and couples said information from said lamps to said processors.

7. A system as in claim **6**, further comprising a plurality of said lighting units, each of said lighting units including a serial communication controller communicating with one of said serial communication controllers of said distribution unit.

8. A method, comprising:

determine a desired position for each of a plurality of movable lamps which are each located in different respective locations; and

send the same information indicative of said desired location to each of said plurality of movable lamps.

9. A method as in claim 8, further comprising calculating desired pointing positions in the lamps from said same information, based on individual location information in the lamps.

10. A method as in claim 9, wherein said individual location information includes information indicative of a lamp position in a same coordinate system as said same information, and information indicative of an orientation of the lamp.

11. A method as in claim 9, wherein said same information is in a Cartesian coordinate system.

12. A method as in claim 8, further comprising sending information indicative of a time of movement, to each of said lamps.

13. A method as in claim 9, further comprising sending information indicative of a time of movement to each of said lamps, and wherein said calculating also comprises calculating an amount of movement to be taken at each of a plurality of time durations.

14. A method as in claim 13, further comprising defining a travel profile for the lamp.

15. A method as in claim 14, further comprising calculating said amount of movement based on said travel profile.

16. A method as in claim 15, wherein said travel profile is a linear travel profile.

17. A method as in claim 15, wherein said travel profile is a sinusoidal travel profile.

18. A method as in claim 8, further comprising, in each of the lamps, converting said same information to pan and tilt information for each of the lamps.

19. A method as in claim 18, wherein said converting includes using individual location information in the lamps to carry out said converting.

20. A method as in claim 19, further comprising an initial setup of each lamp in which each lamp Is sent data indicative of said individual location information including at least its position in a coordinate system and its orientation.

21. A method as in claim 19, wherein said converting comprises using a lookup table to determine values.

22. A method as in claim 19, wherein said converting comprises using a successive approximation calculation to determine values.

23. A method, comprising: storing a information for each of a plurality of lamps in a coordinate system, in a console that controls said each of said plurality of lamps; sending said information to said each of said plurality of lamps in said coordinate system; and in each of said plurality of lamps, using individual information indicative of an individual location of said lamp to convert said information from said coordinate system into information indicative of a pointing direction of each lamp.

24. A method as in claim 23, wherein said individual information includes pan and tilt angles for said each lamp.

25. A method as in claim 23, wherein said individual information includes information unit transformed coordinate system for each of said lamps.

26. A method as in claim 23, wherein said information is a cue which stores information for each of a plurality of lamps using common information for said each of said plurality of lamps.

27. A method as in claim 26, further comprising recalling said cue and storing sending the same information to the each of said plurality of lamps that are respectively in different locations.

28. A method as in claim 23, wherein said information represents a pointing at mode in which the information represents a location where the lamp should point.

29. A method as in claim 23, wherein said information represents a point away mode which indicates where the lamp should point away from.

30. A method as in claim 23, wherein said information represents a point parallel mode in which each of a plurality of lamps are instructed to point in a direction parallel to a specified direction.

31. A method as in claim 29, wherein each of said plurality of lamps are instructed to point parallel to one another.

32. A method, comprising:

providing data indicative of the coordinate system to each of a plurality of remote lamps, wherein each of said remote lamps are at different locations relative to the coordinate system; and

using said data in the remote lamps to cause each of the remote lamps to point parallel to one another.

33. A method as in claim 32, wherein said data also includes defining data indicative of a timing of a lamp movement, and wherein each of said data in said remote lamps carry out said data movement.

34. A method as in claim 33, wherein said defining data includes a time for a sweep, and causes said lamps to carry out parallel sweeps.

35. A method as in claim 33, further comprising providing initialization data to each of said lamps indicative of their individual location, and using said initialization data to interpret said data in said coordinate system.

36. A method as in claim 32, wherein said coordinate system is a Cartesian coordinate system.

37. A method as in claim 34, wherein said lamps calculating from said time of movement how much movement to be carried out in each of a plurality of time durations.

38. A method as in claim 32, wherein said using comprises converting said data in the coordinate system into data indicative of pan and tilt for each of the remote lamps.

39. A method as in claim 34 wherein said defining also comprises defining a travel profile for the lamp.

40. A method as in claim 39, wherein said travel profile is a linear travel profile.

41. A method as in claim 39, wherein said travel profile is a sinusoidal travel profile.

42. A lighting console, which includes a memory, and which stores information in said memory indicative of a desired position of pointing for a plurality of controlled lighting lamps which are in different locations, said desired position of pointing being a single position in a single coordinate system, said lighting console including a user interface portion which enables selection of a specified effect including said desired position, and in response to receiving a control for said specified effect, outputs a signal including said information about said single position in said single coordinate system to each of said plurality of controlled lighting lamps.

43. A console as in claim 42, wherein said coordinate system is a Cartesian coordinate system.

44. A console as in claim 42, wherein said memory also stores time information indicative of a time of travel of movement for said lamps.

45. A console as in claim 42, wherein said coordinates represents a plurality of parallel-pointing lamps.

46. A console as in claim 44, wherein said memory also stores information indicative of a profile for movement of said lamps.

47. A moving lamp system, comprising:

a communication controller, which receives a communication from a remote controller; and

a processing part, which decodes said communication based on information indicative of a specific individual location of the moving lamps, and converts the communication into specific moving instructions for the moving lamp based on said individual location.

48. A system as in claim 47, wherein said communication includes coordinates of an absolute position in a Cartesian coordinate system.

49. A system as in claim 47, wherein said processing part converts information in the coordinate system into pan and tilt angles for the lamp.

50. A system as in claim 47, wherein said processing part also receives a timing signal indicative of a time of movement of said lamp.

51. A system as in claim 50, wherein said processing part causes said lamp to move by a specified amount each time period based on said timing signal.

52. A system as in claim 48, wherein said processing part converts said coordinates of said absolute position into a coordinate system centered on a position of said lamp as indicated by said individual location.

* * * * *

EXHIBIT P

US006891656B2

(12) **United States Patent**
Hunt

(10) **Patent No.:** **US 6,891,656 B2**
(45) **Date of Patent:** **May 10, 2005**

(54) **PIXEL BASED GOBO RECORD CONTROL FORMAT**

(75) Inventor: **Mark Hunt**, Derby (GB)

(73) Assignee: **Production Resource Group, L.L.C.**, New Windsor, NY (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **10/638,124**

(22) Filed: **Aug. 8, 2003**

(65) **Prior Publication Data**

US 2004/0061926 A1 Apr. 1, 2004

Related U.S. Application Data

(63) Continuation of application No. 10/271,521, filed on Oct. 15, 2002, which is a continuation of application No. 09/882,755, filed on Jun. 15, 2001, now Pat. No. 6,466,357, which is a continuation of application No. 09/500,393, filed on Feb. 8, 2000, now Pat. No. 6,256,136, which is a continuation of application No. 09/145,314, filed on Aug. 31, 1998, now Pat. No. 6,057,958.

(60) Provisional application No. 60/059,161, filed on Sep. 17, 1997, and provisional application No. 60/065,133, filed on Nov. 12, 1997.

(51) **Int. Cl.**[7] **G02B 26/00**; G06T 11/20; G06K 9/00; G06K 9/48; G03B 21/00

(52) **U.S. Cl.** **359/291**; 345/441; 345/419; 345/473; 345/604; 345/611; 345/619; 345/764; 382/181; 382/199; 382/192; 382/203; 382/154; 382/167; 353/122; 362/282

(58) **Field of Search** 359/291; 345/431, 345/611, 601, 604, 441, 473, 764, 419, 619, 589; 382/181, 192, 203, 154, 167; 353/15, 25, 122; 362/282, 284, 294

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,891,738 A	*	1/1990	Richardson et al.	362/282
5,113,332 A	*	5/1992	Richardson	362/282
5,282,121 A	*	1/1994	Bornhorst et al.	362/294
5,828,485 A		10/1998	Hewlett	359/291
5,852,443 A		12/1998	Kenworthy	345/431
6,256,136 B1		7/2001	Hunt	359/291
6,486,880 B2		11/2002	Clark-Schreyer et al.	345/473

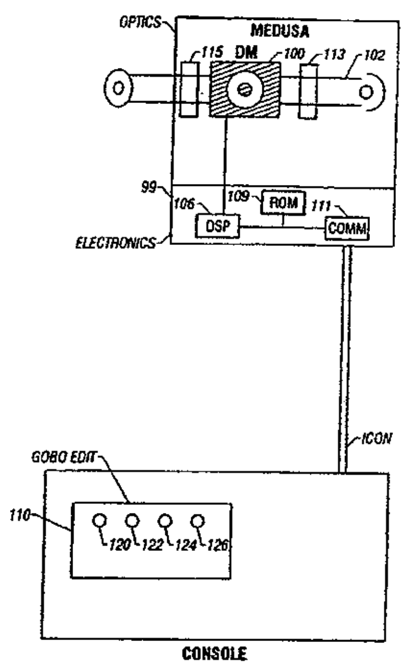
* cited by examiner

Primary Examiner—Loha Ben
(74) *Attorney, Agent, or Firm*—Fish & Richardson P.C.

(57) **ABSTRACT**

A special record format used for commanding light pattern shapes and addressable light pattern shape generator. The command format includes a first part which commands a specified gobo and second parts which command the characteristics of that gobo. The gobo is formed by making a default gobo based on the type and modifying that default gobo to fit the characteristics.

27 Claims, 6 Drawing Sheets



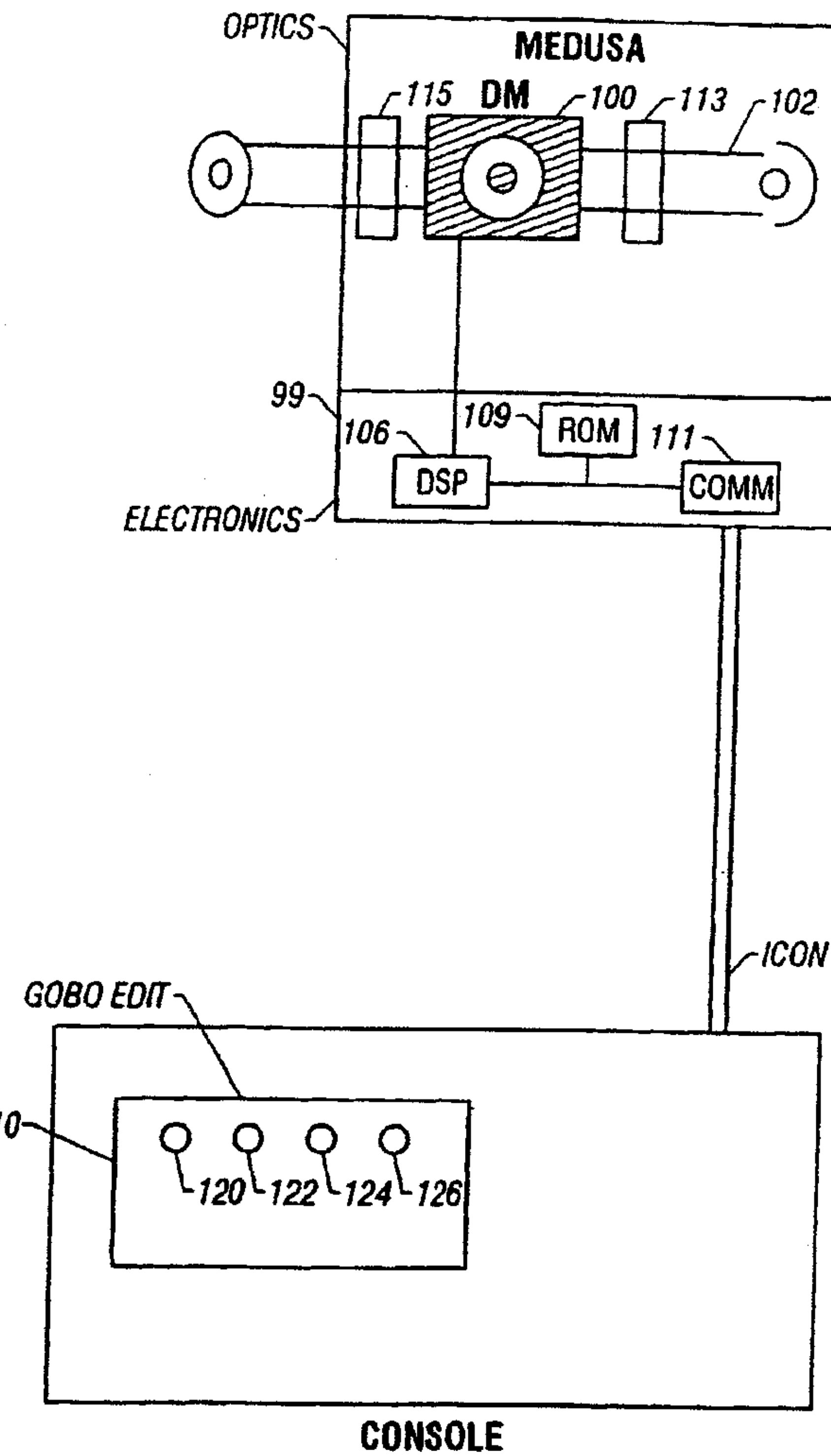

OPTICS
MEDUSA
115
DM
100
113
102
99
106
109
ROM
111
DSP
COMM
ELECTRONICS
ICON
GOBO EDIT
110
120
122
124
126
CONSOLE

FIG. 1

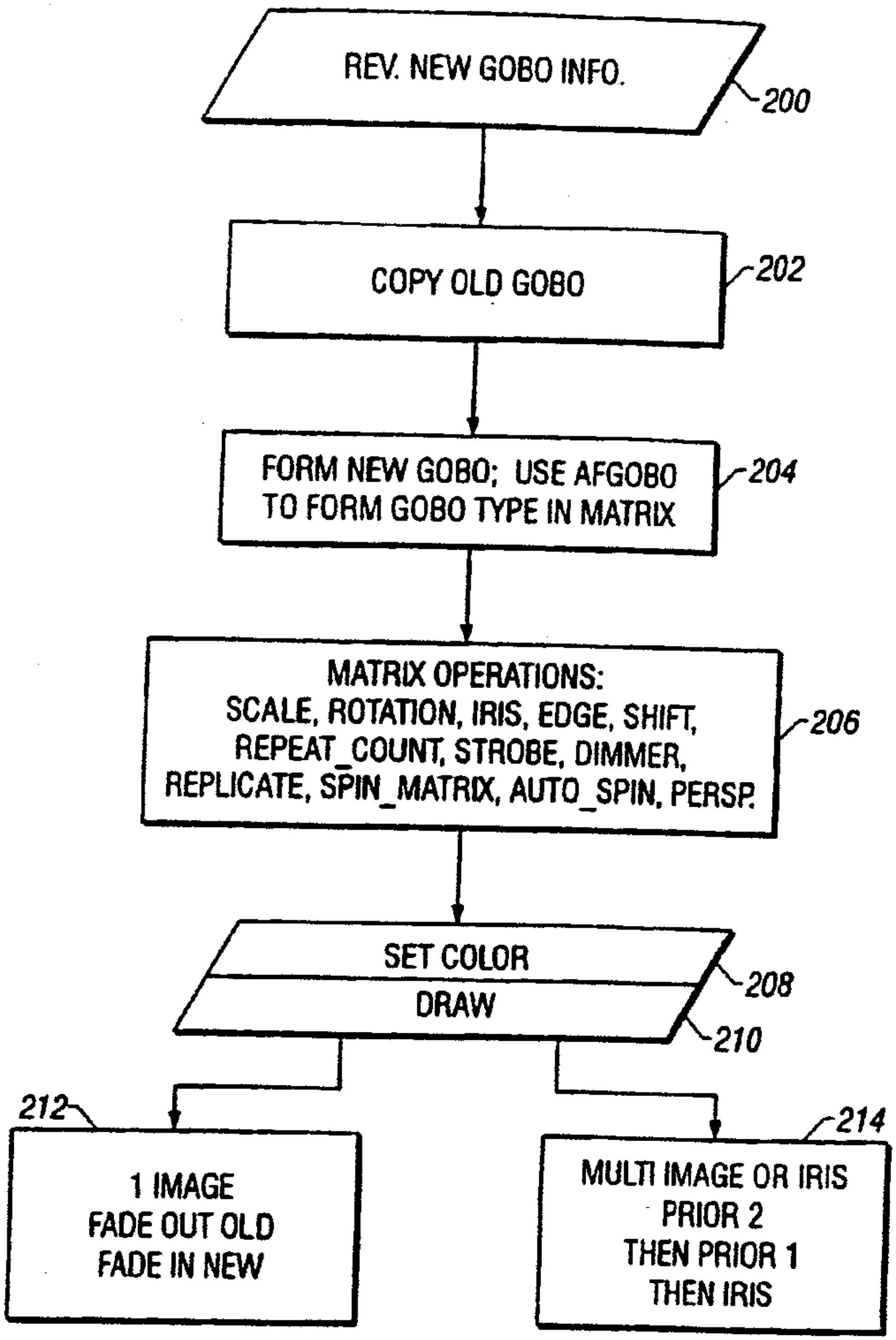

REV. NEW GOBO INFO.
200
COPY OLD GOBO
202
FORM NEW GOBO; USE AFGOBO TO FORM GOBO TYPE IN MATRIX
204
MATRIX OPERATIONS:
SCALE, ROTATION, IRIS, EDGE, SHIFT, REPEAT_COUNT, STROBE, DIMMER, REPLICATE, SPIN_MATRIX, AUTO_SPIN, PERSP.
206
SET COLOR
208
DRAW
210
212
1 IMAGE
FADE OUT OLD
FADE IN NEW
214
MULTI IMAGE OR IRIS
PRIOR 2
THEN PRIOR 1
THEN IRIS

FIG. 2

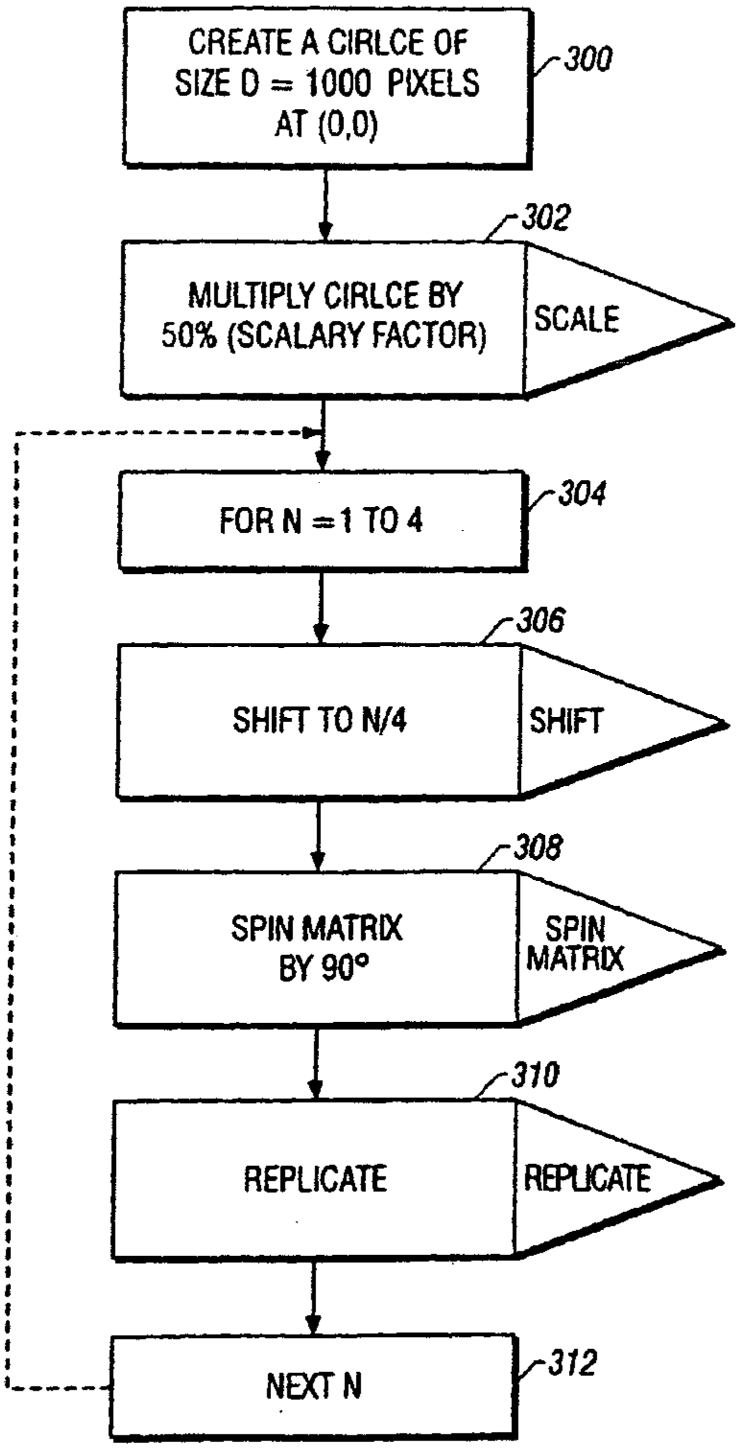

CREATE A CIRLCE OF SIZE D = 1000 PIXELS AT (0,0)
300
302
MULTIPLY CIRLCE BY 50% (SCALARY FACTOR)
SCALE
FOR N =1 TO 4
304
306
SHIFT TO N/4
SHIFT
308
SPIN MATRIX BY 90°
SPIN MATRIX
310
REPLICATE
REPLICATE
NEXT N
312

FIG. 3

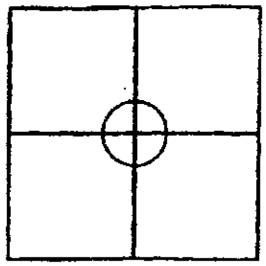


FIG. 4A

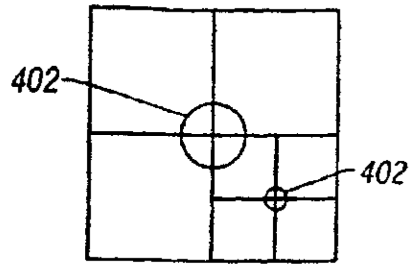


FIG. 4E

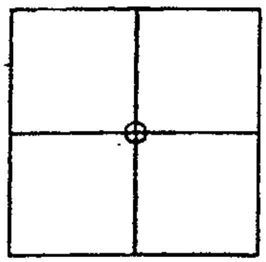


FIG. 4B

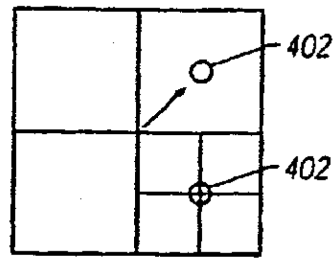


FIG. 4F

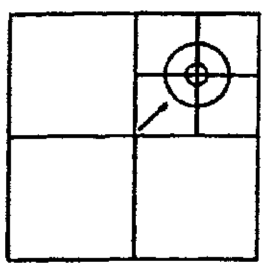


FIG. 4C

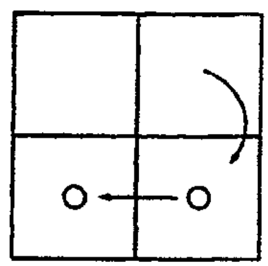


FIG. 4G

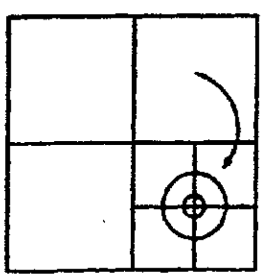


FIG. 4D

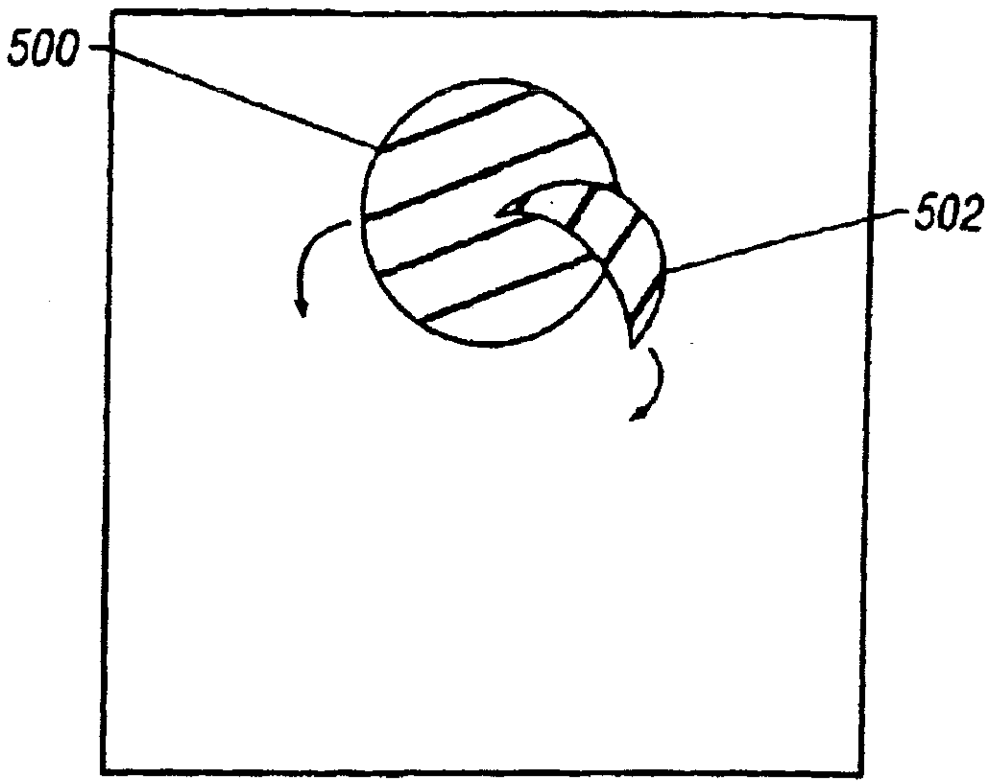


FIG. 5

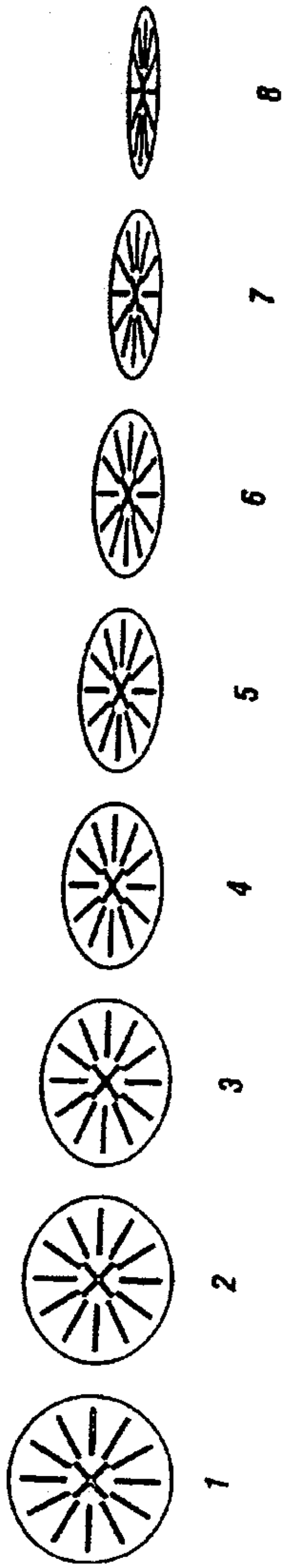


FIG. 6

PIXEL BASED GOBO RECORD CONTROL FORMAT

This application is a continuation of U.S. application Ser. No. 10/271,521, filed Oct. 15, 2002, which is a continuation of U.S. application Ser. No. 09/882,755, filed Jun. 15, 2001 now U.S. Pat. No. 6,466,357, which is a continuation of U.S. application Ser. No. 09/500,393, filed Feb. 8, 2000 now U.S. Pat. 6,256,136, which is a continuation of U.S. application Ser. No. 09/145,314, filed Aug. 31, 1998 now U.S. Pat. No. 6,057,958, which claims priority from U.S. Provisional application Nos. 60/059,161, filed Sep. 17, 1997, and 60/065,133, filed Nov. 12, 1997.

FIELD

The present invention relates to a system of controlling light beam pattern ("gobo") shape in a pixilated gobo control system.

BACKGROUND

Commonly assigned patent application Ser. No. 08/854,353, the disclosure of which is herewith incorporated by reference, describes a stage lighting system which operates based on computer-provided commands to form special effects. One of those effects is control of the shape of a light pattern that is transmitted by the device. This control is carried out on a pixel-by-pixel basis, hence referred to in this specification as pixilated. Control is also carried out using an x-y controllable device. The preferred embodiment describes using a digital mirror device, but other x-y controllable devices such as a grating light valve, are also contemplated.

The computer controlled system includes a digital signal processor **106** which is used to create an image command. That image command controls the pixels of the x-y controllable device to shape the light that it is output from the device.

The system described in the above-referenced application allows unparalleled flexibility in selection of gobo shapes and movement. This opens an entirely new science of controlling gobos. The present inventors found that, unexpectedly, even more flexibility is obtained by a special control language for controlling those movements.

SUMMARY

The present disclosure defines a way of communicating with an x-y controllable device to form special electronic light pattern shapes. More specifically, the present application describes using a control language to communicate with an electronic gobo in order to reposition part or all of the image that is shaping the light.

BRIEF DESCRIPTION OF THE DRAWINGS

These and other aspects of the invention will now be described with reference to the attached drawings, in which:

FIG. **1** shows a block diagram of the basic system operating the embodiment;

FIG. **2** shows a basic flowchart of operation;

FIG. **3** shows a flowchart of forming a replicating circles type gobo;

FIGS. **4A** through **4G** show respective interim results of carrying out the replicating circles operation;

FIG. **5** shows the result of two overlapping gobos rotating in opposite directions; and

FIGS. **6(1)** through **6(8)** show a z-axis flipping gobo.

DESCRIPTION OF THE PREFERRED EMBODIMENT

FIG. **1** shows a block diagram of the hardware used according to the preferred embodiment. As described above, this system uses a digital mirror device **100**, which has also been called a digital mirror device ("DMD") and a digital light processor device ("DLP"). More generally, any system which allows controlling shape of light on a pixel basis, including a grating light valve, could be used as the light shaper. This light shaper forms the shape of light which is transmitted. FIG. **1** shows the light being transmitted as **102**, and shows the transmitted light. The information for the digital mirror **100** is calculated by a digital signal processor **106**. Information is calculated based on local information stored in the lamp, e.g., in ROM **109**, and also in information which is received from the console **104** over the communication link.

The operation is commanded according to a format.

The preferred data format provides 4 bytes for each of color and gobo control information.

The most significant byte of gobo control data, ("dfGobo") indicates the gobo type. Many different gobo types are possible. Once a type is defined, the gobo formed from that type is represented by a number. That type can be edited using a special gobo editor described herein. The gobo editor allows the information to be modified in new ways, and forms new kinds of images and effects.

The images which are used to form the gobos may have variable and/or moving parts. The operator can control certain aspects of these parts from the console via the gobo control information. The type of gobo controls the gobo editor to allow certain parameters to be edited.

The examples given below are only exemplary of the types of gobo shapes that can be controlled, and the controls that are possible when using those gobo shapes. Of course, other controls of other shapes are possible and predictable based on this disclosure.

A first embodiment is the control of an annulus, or "ring" gobo. The DMD **100** in FIG. **1** is shown with the ring gobo being formed on the DMD. The ring gobo is type 000A. When the gobo type 0A is enabled, the gobo editor **110** on the console **104** is enabled and the existing gobo encoders **120**, **122**, **124**, and **126** are used. The gobo editor **110** provides the operator with specialized control over the internal and the external diameters of the annulus, using separate controls in the gobo editor.

The gobo editor and control system also provides other capabilities, including the capability of timed moves between different edited parameters. For example, the ring forming the gobo could be controlled to be thicker. The operation could then effect a timed move between these "preset" ring thicknesses. Control like this cannot even be attempted with conventional fixtures.

Another embodiment is a composite gobo with moving parts. These parts can move though any path that are programmed in the gobo data itself. This is done in response to the variant fields in the gobo control record, again with timing. Multiple parts can be linked to a single control allowing almost unlimited effects.

Another embodiment of this system adapts the effect for an "eye" gobo, where the pupil of the eye changes its position (look left, look right) in response to the control.

Yet another example is a Polygon record which can be used for forming a triangle or some other polygonal shape.

The control can be likened to the slider control under a QuickTime movie window, which allows you to manually move to any point in the movie. However, our controls need not be restricted to timelines.

Even though such moving parts are used, scaling and rotation on the gobo is also possible.

The following type assignments are contemplated:

00–0F=FixedGobo (with no "moving parts")
10–1F=SingleCntrl (with 1 "moving part")
20–2F=DoubleCntrl (with 2 "moving parts")
30–FF=undefined, reserved.

The remaining control record bytes for each type are defined as follows:

Byte: memory	dfGobo2	dfGobo3	dfGobo4	#gobos/ type,	total
FixedGobo:	ID[23:16]	ID[15:8]	ID[7:0]	16 M/type	256 M
SingleCntrl:	ID[15:8]	ID[7:0]	control#1	64 k/type	1 M
DoubleCntrl:	ID[7:0]	control#2	control#1	256/type	4 k

As can be seen from this example, this use of the control record to carry control values does restrict the number of gobos which can be defined of that type, especially for the 2-control type.

Console Support:

The use of variant part gobos requires no modifications to existing console software for the ICON (7M) console. The Gobo editor in current ICON software already provides 4 separate encoders for each gobo. These translate directly to the values of the 4 bytes sent in the communications data packet as follows:

Byte: dfGobo Enc: TopRight	dfGobo2 MidRight	dfGobo3 BotRight	dfGobo4 BotLeft
FixedGobo:	ID[23:16]	ID[15:8]	ID[7:0]
SingleCntrl:	ID[15:8]	ID[7 0]	control#1
DoubleCntrl:	ID[7:0]	control#2	control#1

These values would be part of a preset gobo, which could be copied as the starting point.

Once these values are set, the third and fourth channels automatically become the inner/outer radius controls. Using two radii allows the annulus to be turned "inside out".

Each control channel's data always has the same meaning within the console. The console treats these values as simply numbers that are passed on. The meanings of those numbers, as interpreted by the lamps change according to the value in dfGobo.

The lamp will always receives all 4 bytes of the gobo data in the same packet. Therefore, a "DoubleCntrl" gobo will always have the correct control values packed along with it.

Hence, the console needs no real modification. If a "soft" console is used, then name reassignments and/or key reassignments may be desirable.

Timing:

For each data packet, there is an associated "Time" for gobo response. This is conventionally taken as the time allotted to place the new gobo in the light gate. This delay has been caused by motor timing. In this system, variant gobo, the control is more dynamically used. If the non-variant parts of the gobo remain the same, then it is still the same gobo, only with control changes. Then, the time value is interpreted as the time allowed for the control change.

Since different gobo presets (in the console) can reference the same gobo, but with different control settings, this allows easily programmed timed moves between different annuli, etc.

Internal Workings:

When the gobo command data is extracted from the packet at the lamp, the dfGobo byte is inspected first, to see if either dfGobo3 or dfGobo4 are significant in selecting the image. In the case of the "Cntrl" variants, one or both of these bytes is masked out, and the resulting 32-bit number is used to search for a matching gobo image (by Gobo_ID) in the library stored in the lamp's ROM **109**.

If a matching image is found, and the image is not already in use, then the following steps are taken:

1) The image data is copied into RAM, so that its fields may be modified by the control values. This step will be skipped if the image is currently active.

2) The initial control values are then recovered from the data packet, and used to modify certain fields of the image data, according to the control records.

3) The image is drawn on the display device, using the newly-modified fields in the image data.

If the image is already in use, then the RAM copy is not altered. Instead, a time-sliced task is set up to slew from the existing control values to those in the new data packet, in a time determined by the new data packet.

At each vertical retrace of the display, new control values are computed, and steps 2 (using the new control values) and 3 above are repeated, so that the image appears modified with time.

The Image Data Records:

All images stored in the lamp are in a variant record format:

Header:	
Length 32 bits, offset to next gobo in list.	
Gobo _ID 32 bits, serial number of gobo.	
Gobo records:	
Length	32 bits, offset to next record.
Opcode	16 bits, type of object to be drawn.
Data	Variant part - data describing object.
_Length	32 bits, offset to next record.
Opcode	16 bits, type of object to be drawn.
Data	Variant part - data describing object.
_ EndMarker 64 bits, all zeroes - indicates end of gobo data.	
+ Next gobo, or End Marker, indicating end of gobo list.	

Gobos with controls are exactly the same, except that they contain control records, which describe how the control values are to affect the gobo data. Each control record contains the usual length and Opcode fields, and a field containing the control number (1 or 2).

These are followed by a list of "field modification" records. Each record contains information about the offset (from the start of the gobo data) of the field, the size (8, 16 or 32 bits) of the field, and how its value depends on the control value.

Length	32 bits, offset to next record
Opcode	16 bits = control_record (constant)

-continued

CntrlNum	16 bits = 1 or 2 (control number) /* field modification record #1 */
Address	16 bits, offset from start of gobo to affected field.
Flags	16 bits, information about field (size, signed, etc)
Scale	16 bits, scale factor applied to control before use
ZPoint	16 bits, added to control value after scaling. /* field modification record #2 */
Address	16 bits, offset from start of gobo to affected field.
Flags	16 bits, information about field (size, signed, etc)
Scale	16 bits, scale factor applied to control before use
ZPoint	16 bits, added to control value after scaling.

As can be seen, a single control can have almost unlimited effects on the gobo, since ANY values in the data can be modified in any way, and the number of field modification records is almost unlimited.

Note that since the control records are part of the gobo data itself, they can have intimate knowledge of the gobo structure. This makes the hard-coding of field offsets acceptable.

In cases where the power offered by this simple structure is not sufficient, a control record could be defined which contains code to be executed by the processor. This code would be passed parameters, such as the address of the gobo data, and the value of the control being adjusted.

EXAMPLE RECORDS

The Annulus record has the following format:

Length	32 bits
Opcode	16 bits, = type_annulus
Pad	16 bits, unused
Centre_x	16 bits, x coordinate of centre
Centre_y	16 bits, y coordinate of centre
OuterRad	16 bits, outside radius (the radii get swapped when drawn if their values are in the wrong order)
InnerRad	16 bits, inside radius

It can be seen from this that it is easy to "target" one of the radius parameters from a control record. Use of two control records, each with one of the radii as a target, would provide full control over the annulus shape.

Note that if the centre point coordinates are modified, the annulus will move around the display area, independent of any other drawing elements in the same gobo's data.

The Polygon record for a triangle has this format:

Length	32 bits
Opcode	16 bits, = type_polygon
Pad	16 bits, vertex count = 3
Centre_x	16 bits, x coordinate of vertex
Centre_y	16 bits, y coordinate of vertex
Centre_x	16 bits, x coordinate of vertex
Centre_y	16 bits, y coordinate of vertex
Centre_x	16 bits, x coordinate of vertex
Centre_y	16 bits, y coordinate of vertex

It is easy to modify any of the vertex coordinates, producing distortion of the triangle.

The gobo data can contain commands to modify the drawing environment, by rotation, scaling, offset, and color control, the power of the control records is limitless.

Second Embodiment

This second embodiment provides further detail about implementation once the gobo information is received.

Gobo information is, at times, being continuously calculated by DSP **106**. The flowchart of FIG. **2** shows the handling operation that is carried out when new gobo information is received.

At step **200**, the system receives new gobo information. In the preferred embodiment, this is done by using a communications device ill in the lamp **99**. The communications device is a mailbox which indicates when new mail is received. Hence, the new gobo information is received at step **200** by determining that new mail has been received.

At step **202**, the system copies the old gobo and switches pointers. The operation continues using the old gobo until the draw routine is called later on.

At step **204**, the new information is used to form a new gobo. The system uses a defined gobo ("dfGobo") as discussed previously which has a defined matrix. The type dfGobo is used to read the contents from the memory **109** and thereby form a default image. That default image is formed in a matrix. For example, in the case of an annulus; a default size annulus can be formed at position **0,0** in the matrix. An example of forming filled balls is provided herein.

Step **206** represents calls to subroutines. The default gobo is in the matrix, but the power of this system is its ability to very easily change the characteristics of that default gobo. In this embodiment, the characteristics are changed by changing the characteristics of the matrix and hence, shifting that default gobo in different ways. The matrix operations, which are described in further detail herein, include scaling the gobo, rotation, iris, edge, strobe, and dimmer. Other matrix operations are possible. Each of these matrix operations takes the default gobo, and does something to it.

For example, scale changes the size of the default gobo rotation rotates the default gobo by a certain amount.

Iris simulates an iris operation by choosing an area of interest, typically circular, and erasing everything outside that area of interest. This is very easily done in the matrix, since it simply defines a portion in the matrix where all black is written.

Edge effects carry out certain effects on the edge is such as softening the edge. This determines a predetermined thickness, which is translated to a predetermined number of pixels, and carries out a predetermined operation on the number of pixels. For example, for a 50% edge softening, every other pixel can be turned off. The strobe is in effect that allows all pixels to be turned on and off at a predetermined frequency, i.e., 3 to 10 times a second. The dimmer allows the image to be made dimmer by turning off some of the pixels at predetermined times.

The replicate command forms another default gobo, to allow two different gobos to be handled by the same record. This will be shown with reference to the exemplary third embodiment showing balls. Each of those gobos is then handled as the same unit and the entirety of the gobos can be, for example, rotated. The result of step **206** and all of these subroutines that are called is that the matrix includes information about the bits to be mapped to the digital mirror **100**.

At step **208**, the system then obtains the color of the gobos from the control record discussed previously. This gobo color is used to set the appropriate color changing circuitry

113 and 115 in the lamp 99. Note that the color changing circuitry is shown both before and after the digital mirror 100. It should be understood that either of those color changing circuits could be used by itself.

At step 210, the system calls the draw routine in which the matrix is mapped to the digital mirror. This is done in different ways depending on the number of images being used. Step 212 shows the draw routine for a single image being used as the gobo. In that case, the old gobo, now copied as shown in step 202, is faded out while the new gobo newly calculated is faded in. Pointers are again changed so that the system points to the new gobo. Hence, this has the effect of automatically fading out the old gobo and fading in the new gobo.

Step 214 schematically shows the draw routine for a system with multiple images for an iris. In that system, one of the gobos is given priority over the other. If one is brighter than the other, then that one is automatically given priority. The one with priority 2, the lower priority one, is written first. Then the higher priority gobo is written. Finally, the iris is written which is essentially drawing black around the edges of the screen defined by the iris. Note that unlike a conventional iris, this iris can take on many different shapes. The iris can take on not just a circular shape, but also an elliptical shape, a rectangular shape, or a polygonal shape. In addition, the iris can rotate when it is non-circular so that for the example of a square iris, the edges of the square can actually rotate.

Returning to step 206, in the case of a replicate, there are multiple gobos in the matrix. This allows the option of spinning the entire matrix, shown as spin matrix.

An example will now be described with reference to the case of repeating circles. At step 200, the new gobo information is received indicating a circle. This is followed by the other steps of 202 where the old gobo is copied, and 204 where the new gobo is formed. The specific operation forms a new gobo at step 300 by creating a circle of size diameter equals 1000 pixels at origin 00. This default circle is automatically created. FIG. 4A shows the default gobo which is created, a default size circle at 00. It is assumed for purposes of this operation that all of the circles will be the same size.

At step 302, the circle is scaled by multiplying the entire circle by an appropriate scaling factor. Here, for simplicity, we are assuming a scaling factor of 50% to create a smaller circle. The result is shown in FIG. 4B. A gobo half the size of the gobo of FIG. 4A is still at the origin. This is actually the scale of the subroutine as shown in the right portion of step 302. Next, since there will be four repeated gobos in this example, a four-loop is formed to form each of the gobos at step 304. Each of the gobos is shifted in position by calling the matrix operator shift. In this example, the gobo is shifted to a quadrant to the upper right of the origin. This position is referred to as π over 4 in the FIG. 3 flowchart and results in the gobo being shifted to the center portion of the top right quadrant as shown in FIG. 4C. This is again easily accomplished within the matrix by moving the appropriate values. At step 308, the matrix is spun by 90 degrees in order to put the gobo in the next quadrant as shown in FIG. 4D in preparation for the new gobo being formed into the same quadrant. Now the system is ready for the next gobo, thereby calling the replicate command which quite easily creates another default gobo circle and scales it. The four-loop is then continued at step 312.

The replicate process is shown in FIG. 4E where a new gobo 402 is formed in addition to the existing gobo 400. The system then passes again through the four-loop, with the

results being shown in the following figures. In FIG. 4F, the new gobo 402 is again moved to the upper right quadrant (step 306). In FIG. 4G, the matrix is again rotated to leave room for a new gobo in the upper right quadrant. This continues until the end of the four-loop. Hence, this allows each of the gobos to be formed.

Since all of this is done in matrix operation, it is easily programmable into the digital signal processor. While the above has given the example of a circle, it should be understood that this scaling and moving operation can be carried out for anything. The polygons, circles, annulus, and everything else is easily scaled.

The same operation can be carried out with the multiple parameter gobos. For example, for the case of a ring, the variable takes the form annulus (inner R, outer R, x and y). This defines the annulus and turns of the inner radius, the outer radius, and x and y offsets from the origin. Again, as shown in step 3, the annulus is first written into the matrix as a default size, and then appropriately scaled and shifted. In terms of the previously described control, the ring gobo has two controls: control 1 and control 2 defined the inner and outer radius.

Each of these operations is also automatically carried out by the command repeat count which allows easily forming the multiple position gobo of FIGS. 4A–4G. The variable auto spin defines a continuous spin operation. The spin operation commands the digital signal processor to continuously spin the entire matrix by a certain amount each time.

One particularly interesting feature available from the digital mirror device is the ability to use multiple gobos which can operate totally separately from one another raises the ability to have different gobos spinning in different directions. When the gobos overlap, the processor can also calculate relative brightness of the two gobos. In addition, one gobo can be brighter than the other. This raises the possibility of a system such as shown in FIG. 5. Two gobos are shown spinning in opposite directions: the circle gobo 500 is spinning the counterclockwise direction, while the half moon gobo 502 is spinning in the clockwise direction. At the overlap, the half moon gobo which is brighter than the circle gobo, is visible over the circle gobo. Such effects were simply not possible with previous systems. Any matrix operation is possible, and only a few of those matrix operations have been described herein.

A final matrix operation to be described is the perspective transformation. This defines rotation of the gobo in the Z axis and hence allows adding depth and perspective to the gobo. For each gobo for which rotation is desired, a calculation is preferably made in advance as to what the gobo will look like during the Z axis transformation. For example, when the gobo is flipping in the Z axis, the top goes back and looks smaller while the front comes forward and looks larger. FIGS. 6(1)–6(8) show the varying stages of the gobo flipping.

In FIG. 6(8), the gobo has its edge toward the user. This is shown in FIG. 6(8) as a very thin line, e.g., three pixels wide, although the gobo could be zero thickness at this point.

Automatic algorithms are available for such Z axis transformation, or alternatively a specific Z axis transformation can be drawn and digitized automatically to enable a custom look.

Although only a few embodiments have been described in detail above, other embodiments are contemplated by the inventor and are intended to be encompassed within the following claims. In addition, other modifications are contemplated and are also intended to be covered.

What is claimed is:

1. A system, comprising:

a console, having at least one file therein which is indicative of an image, at least one control therein which controls some aspect of a light pattern shape based on the image, and a processing element that produces an output signal defining a record that includes information about said light pattern shape and at least one parameter representing said light pattern shape.

2. A system as in claim 1, wherein said console includes a digital signal processor which computes information related to said output signal.

3. A system as in claim 1, wherein said control allows controlling a size of the light pattern shape.

4. A system as in claim 1, wherein said control allows controlling e basic shape and editing the shape.

5. A system as in claim 1, wherein said control allows controlling a color of the shaped light.

6. A system as in claim 1, wherein said record defines an output signal that controls a light.

7. A system as in claim 1, wherein said record includes a first portion defining a gobo type, and a second portion defining characteristics of said gobo type.

8. A system as in claim 7, wherein said characteristics include size of the gobo as defined by the first portion.

9. A system, comprising:

a console, having at least one control thereon, which stores images, allows selection of one of said images, to define a gobo, and includes a processing element which produces an output signal indicative of the gobo defined by said one of said images.

10. A system as in claim 9, further comprising a control on said console which allows controlling variation of the gobo, and which changes the output signal indicative of said control.

11. A system as in claim 10, wherein the output signal includes a first value indicative of the gobo type, and a second value indicative of said variation.

12. A system as in claim 9, wherein said output signal includes a first value indicative of the gobo type and a second value indicative of additional aspects of the gobo.

13. A system as in claim 12, wherein said first value is a number indicative of a gobo shape.

14. A system as in claim 13, wherein said second value is information indicative of gobo sides.

15. A system as in claim 13, wherein said second value is information indicative of gobo time.

16. A format, comprising, a first set of bits defining the gobo control data which represents a form of a digital gobo and a second set of bits defining color data which represents a color of the digital gobo.

17. A format as in claim 16, wherein said format includes another set of bits indicative of gobo size.

18. A format as in claim 16, wherein said form of the digital gobo includes information indicative of the shape.

19. A format as in claim 16, wherein said first set of bits includes a type assignment which indicates a number of moving parts.

20. A format comprising a first set of bits defining a gobo to be projected, and a second set of bits defining the timing of moving parts of the gobo.

21. A format as in claim 20, wherein said first set of bits defines at least a shape of the gobo.

22. A format as in claim 20, wherein said second set of bits defines timing of a movement of the gobo.

23. A format as in claim 20, wherein said second set of bits defines timing of a change in size of said gobo.

24. A format comprising a first set of bite defining a gobo to be projected, said first set of bits defining a format of a notification of said gobo, and at least one additional set of bits, whose significance is determined by the first set of bits.

25. A format as in claim 24, wherein the first set of bits defines a gobo type and includes a number of parameters of the gobo, and said at least one additional set of bits defines said parameters.

26. A method, comprising:

receiving, in a light projecting element, a signal representing a digital gobo, and

using said signal representing the digital gobo to alter the shape of projected light on a pixel by pixel basis.

27. A method as in claim 26, wherein said using comprises carrying out a matrix arithmetic operation using said signal.

* * * * *

EXHIBIT Q

US006894443B2

(12) **United States Patent**
Hunt et al.

(10) **Patent No.:** **US 6,894,443 B2**
(45) **Date of Patent:** ***May 17, 2005**

(54) **STAGE LIGHTING LAMP UNIT AND STAGE LIGHTING SYSTEM INCLUDING SUCH UNIT**

(75) Inventors: **Mark A. Hunt**, Derby (GB); **Keith J. Owen**, Birmingham (GB); **Michael D. Hughes**, Wolverhampton (GB)

(73) Assignee: **Production Resource Group L.L.C.**, New Windsor, NY (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

This patent is subject to a terminal disclaimer.

(21) Appl. No.: **10/625,465**

(22) Filed: **Jul. 22, 2003**

(65) **Prior Publication Data**

US 2004/0125602 A1 Jul. 1, 2004

Related U.S. Application Data

(63) Continuation of application No. 10/007,008, filed on Dec. 4, 2001, now Pat. No. 6,597,132, which is a continuation of application No. 09/313,418, filed on May 17, 1999, now Pat. No. 6,326,741, which is a continuation of application No. 08/994,036, filed on Dec. 18, 1997, now Pat. No. 5,921,659, which is a division of application No. 08/576,211, filed on Dec. 21, 1995, now Pat. No. 5,788,365, which is a continuation of application No. 08/077,877, filed on Jun. 18, 1993, now Pat. No. 5,502,627.

(30) **Foreign Application Priority Data**

Sep. 25, 1992	(GB)	9220303
Sep. 25, 1992	(GB)	9220309
Apr. 20, 1993	(GB)	9308071

(51) **Int. Cl.**[7] **H05B 37/00**
(52) **U.S. Cl.** **315/312**; 315/316; 362/233
(58) **Field of Search** 315/316, 312, 315/294, 292, 318, 319; 362/301, 233, 293

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,392,187	A		7/1983	Bornhorst	362/233
4,716,344	A		12/1987	Newell et al.	315/294
4,837,665	A	*	6/1989	Hoyer et al.	362/96
4,980,806	A		12/1990	Taylor et al.	315/316

(Continued)

FOREIGN PATENT DOCUMENTS

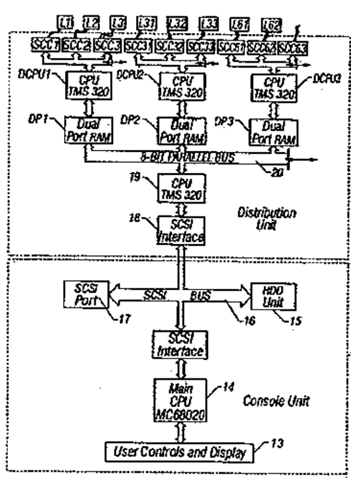
EP	0 495 305	7/1992
FR	2 643 134	8/1990
GB	2 209 229	5/1989
GB	2 231 138	11/1990
WO	89 054 21	6/1989

Primary Examiner—Tuyet Vo
Assistant Examiner—Ephrem Alemu
(74) *Attorney, Agent, or Firm*—Fish & Richardson P.C.

(57) **ABSTRACT**

A stage lighting lamp unit includes a processor for receiving control data from a remote console. Beam orientation data for the lamp unit is passed to the lamp in the form of the x, y and z co-ordinates of a point in space through which the beam is to pass. The processor divides the required lamp travel into a number of stages dependent on execution duration data sent with the position data, and calculates, for each stage, a new value for pan and tilt angles for the lamp. These values are passed to pan and tilt controlling co-processors which control servo-motors for pan and tilt operation. The lamp unit also incorporates a rotatable shutter for interrupting the lamp beam when required. The shutters of all the lamps in a system can be instructed from the remote console to open and close in synchronism, thereby providing a stroboscopic effect.

37 Claims, 10 Drawing Sheets



U.S. PATENT DOCUMENTS

5,005,162 A	4/1991	Mitsumori et al.	369/112.29
5,010,459 A	4/1991	Taylor et al.	315/292
5,045,983 A	9/1991	Shields	367/293
5,414,328 A	5/1995	Hunt et al.	315/292
5,502,627 A	3/1996	Hunt et al.	362/286
5,588,021 A	12/1996	Hunt et al.	370/246
5,769,531 A	6/1998	Hunt et al.	362/233
5,788,365 A	8/1998	Hunt et al.	362/301
5,921,659 A	7/1999	Hunt et al.	362/233
6,326,741 B1	12/2001	Hunt et al.	315/316

* cited by examiner

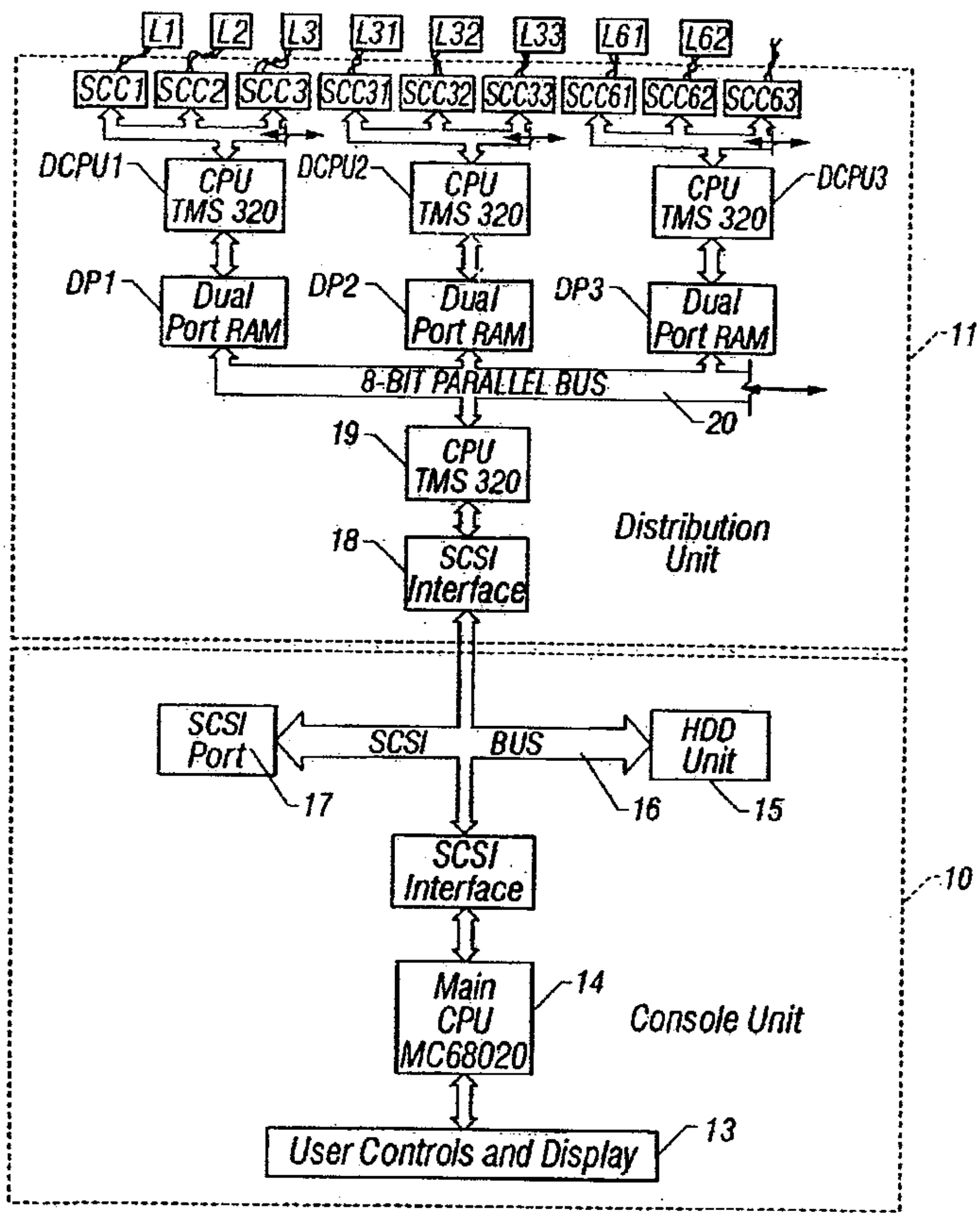

L1
L2
L3
L31
L32
L33
L61
L62
SCC1
SCC2
SCC3
SCC31
SCC32
SCC33
SCC61
SCC62
SCC63
DCPU1
CPU TMS 320
DCPU2
CPU TMS 320
CPU TMS 320
DCPU3
DP1
Dual Port RAM
DP2
Dual Port RAM
DP3
Dual Port RAM
11
8-BIT PARALLEL BUS
20
19
CPU TMS 320
18
SCSI Interface
Distribution Unit
SCSI Port
SCSI BUS
HDD Unit
17
16
15
SCSI Interface
10
Main CPU MC68020
14
Console Unit
User Controls and Display
13

FIG. 1

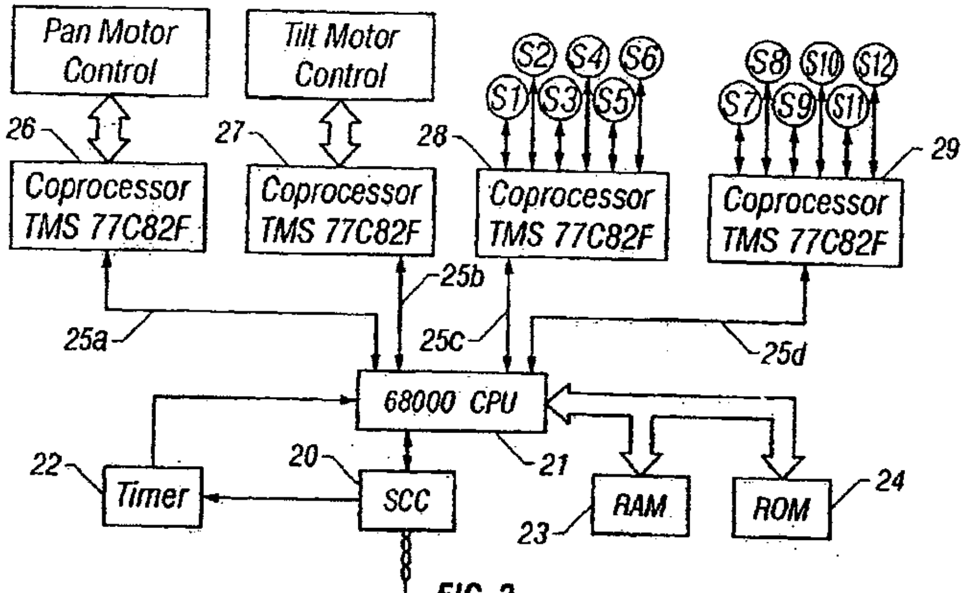

Pan Motor Control
Tilt Motor Control
26
27
28
29
S1
S2
S3
S4
S5
S6
S7
S8
S9
S10
S11
S12
Coprocessor TMS 77C82F
Coprocessor TMS 77C82F
Coprocessor TMS 77C82F
Coprocessor TMS 77C82F
25a
25b
25c
25d
68000 CPU
21
22
Timer
20
SCC
RAM
23
ROM
24

FIG. 2

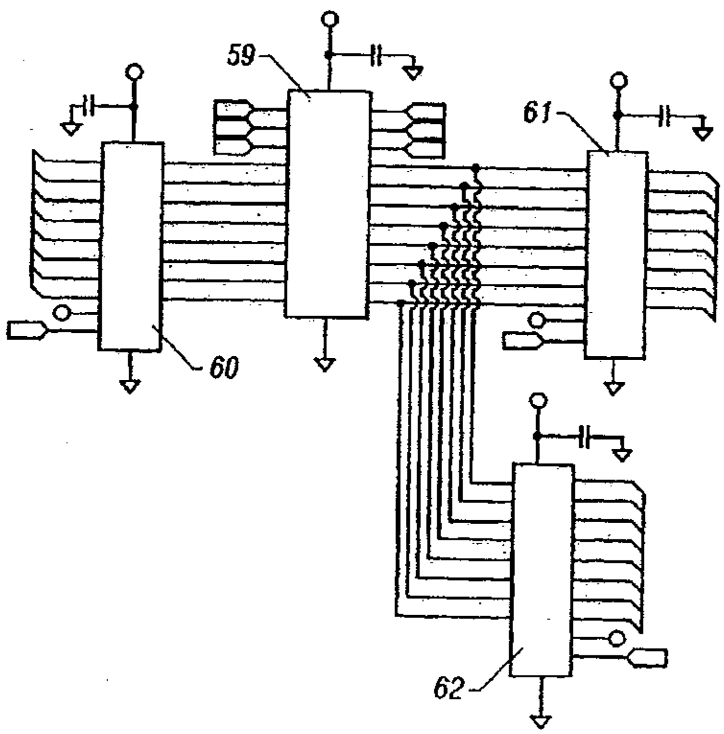

59
61
60
62

FIG. 7

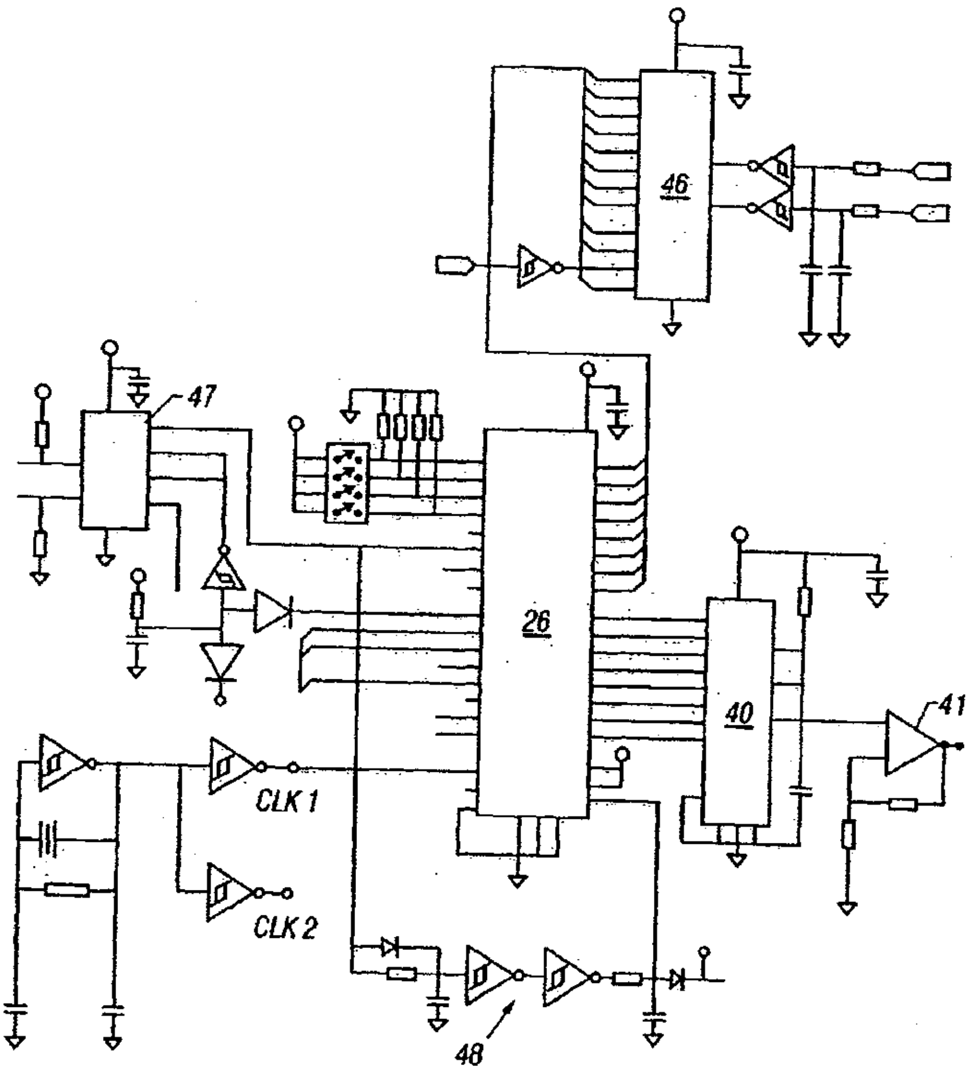


FIG. 3

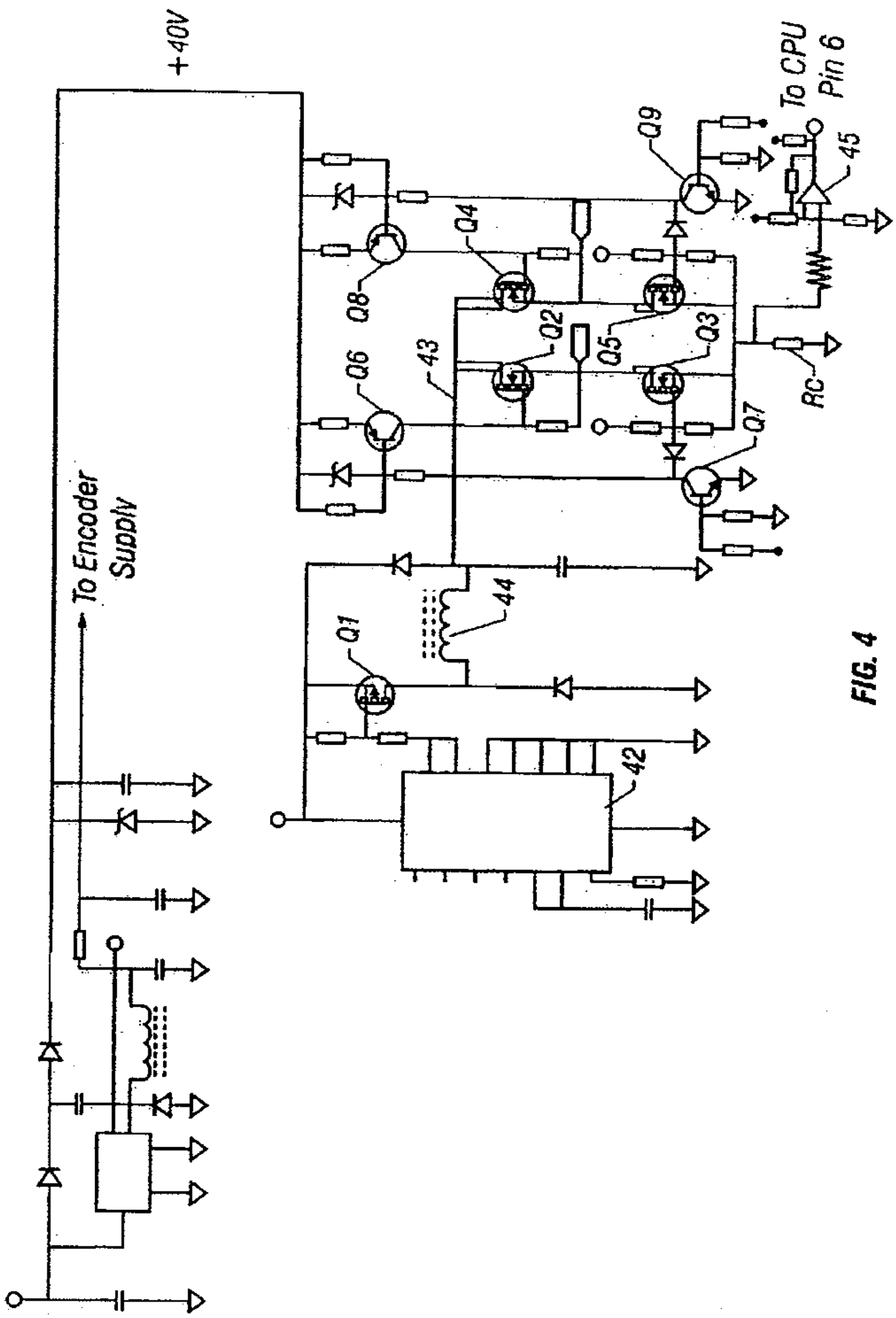

+40V
To CPU
Pin 6
Q9
45
Q4
Q8
Q2
Q5
Q3
Q6
43
Rc
Q7
To Encoder
Supply
44
Q1
42

FIG. 4

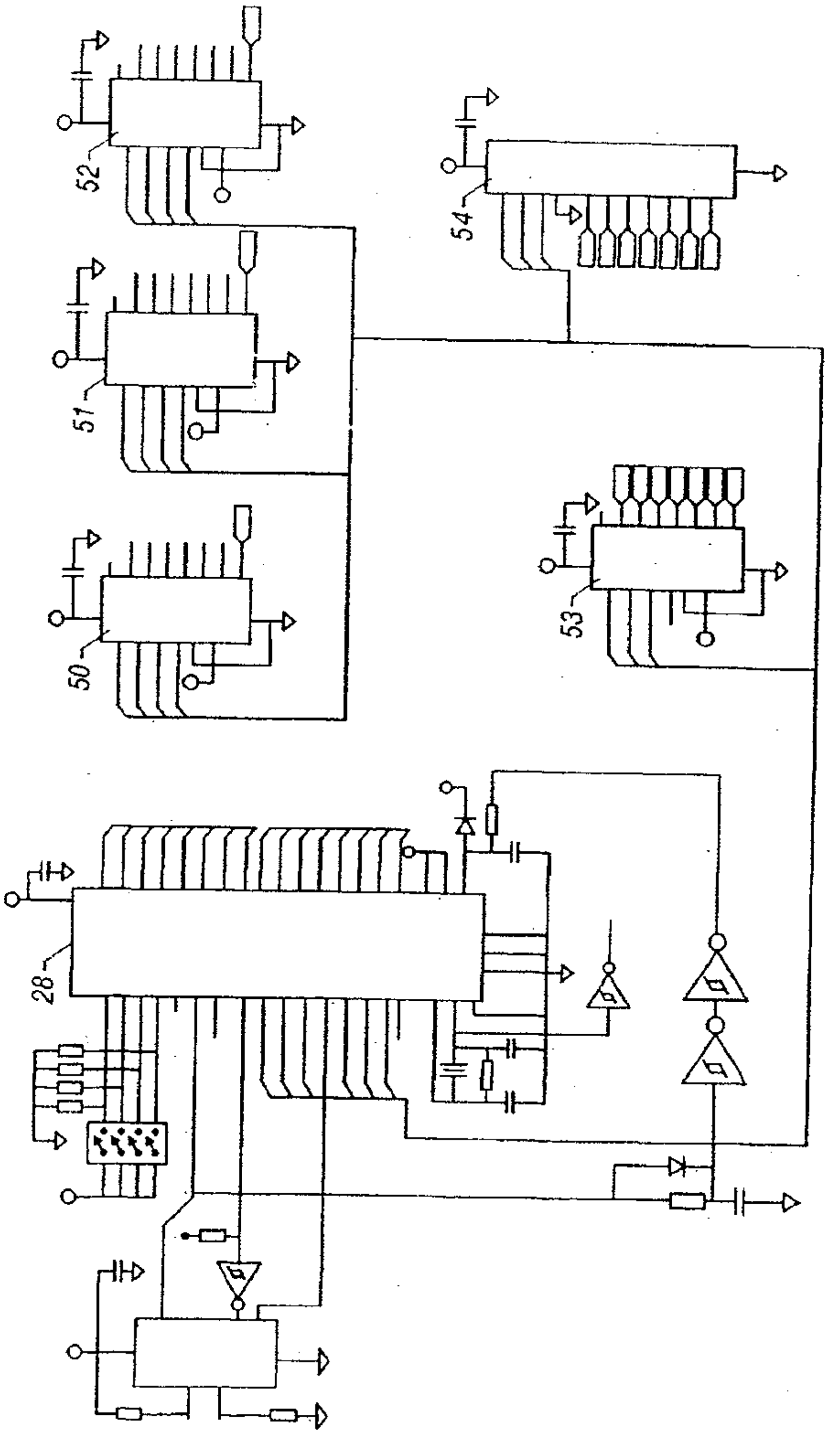


FIG. 5

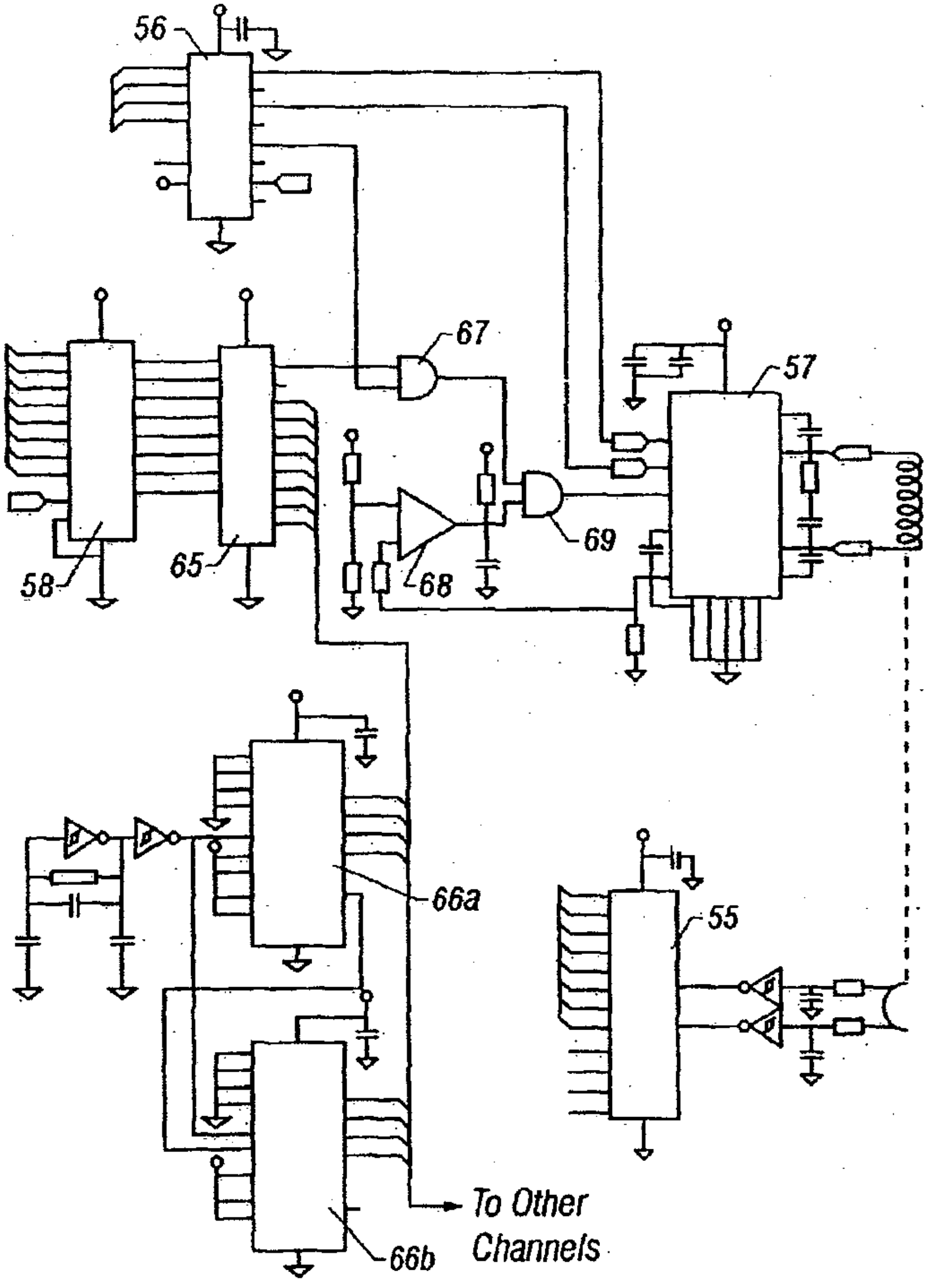


FIG. 6

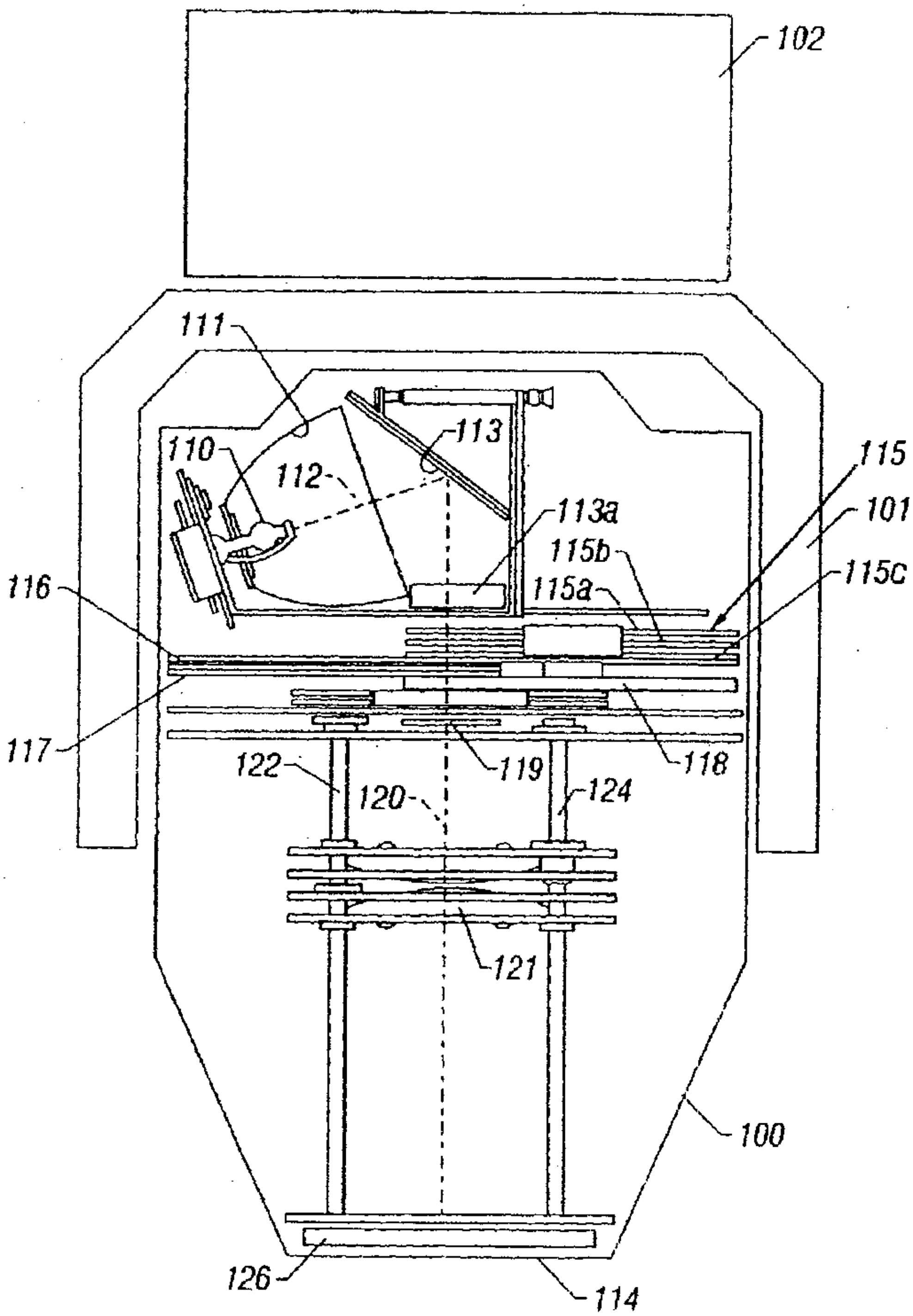

102
111
110
112
113
113a
115
101
115b
115a
115c
116
117
122
120
119
118
124
121
100
126
114

FIG. 8

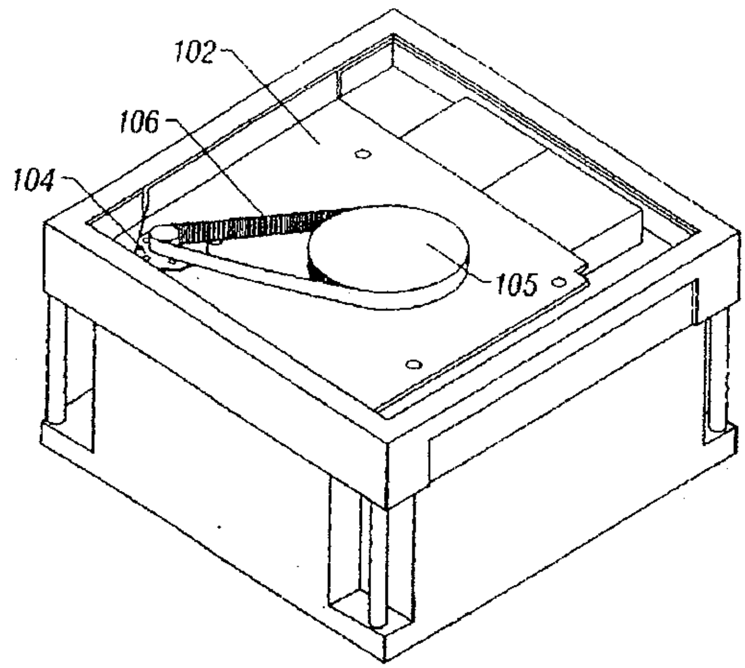


FIG. 9

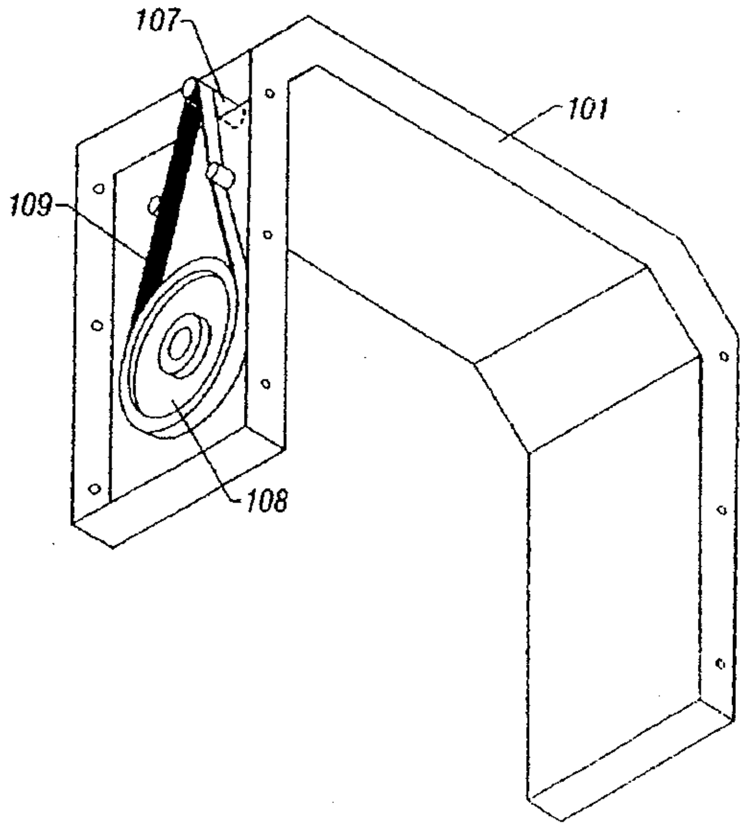
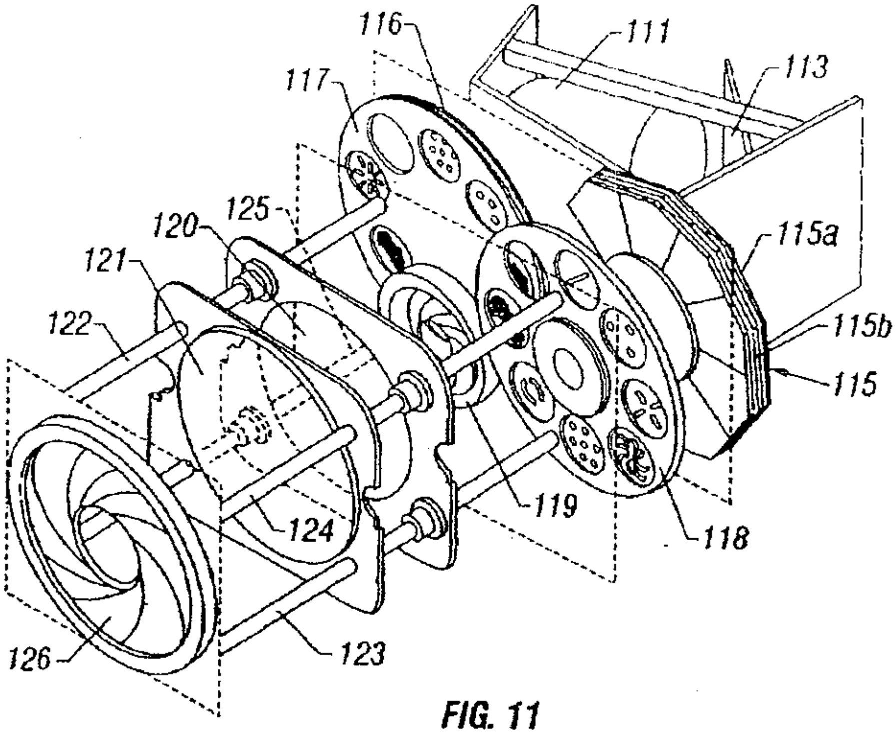
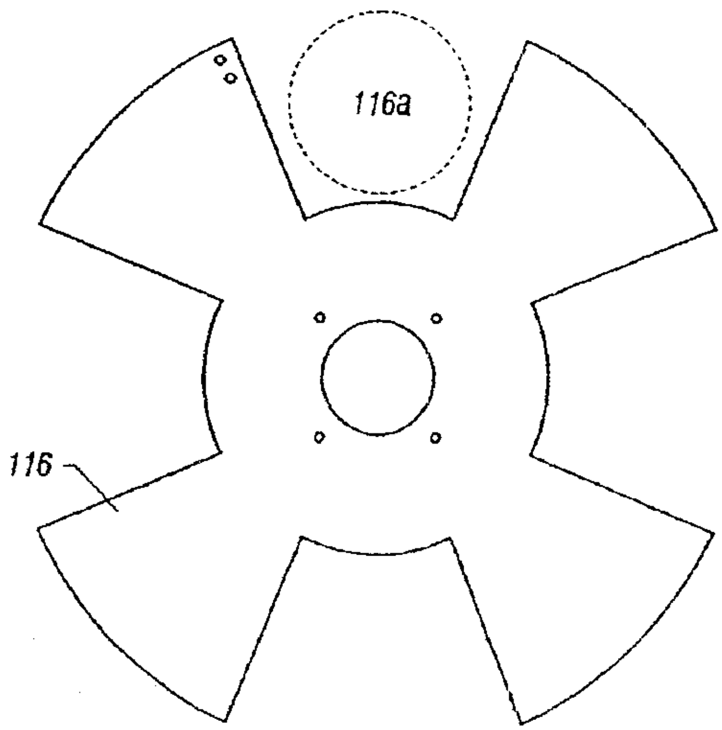


FIG. 10



116
111
113
117
125
120
121
115a
122
115b
115
119
124
118
126
123

FIG. 11



116a
116

FIG. 13

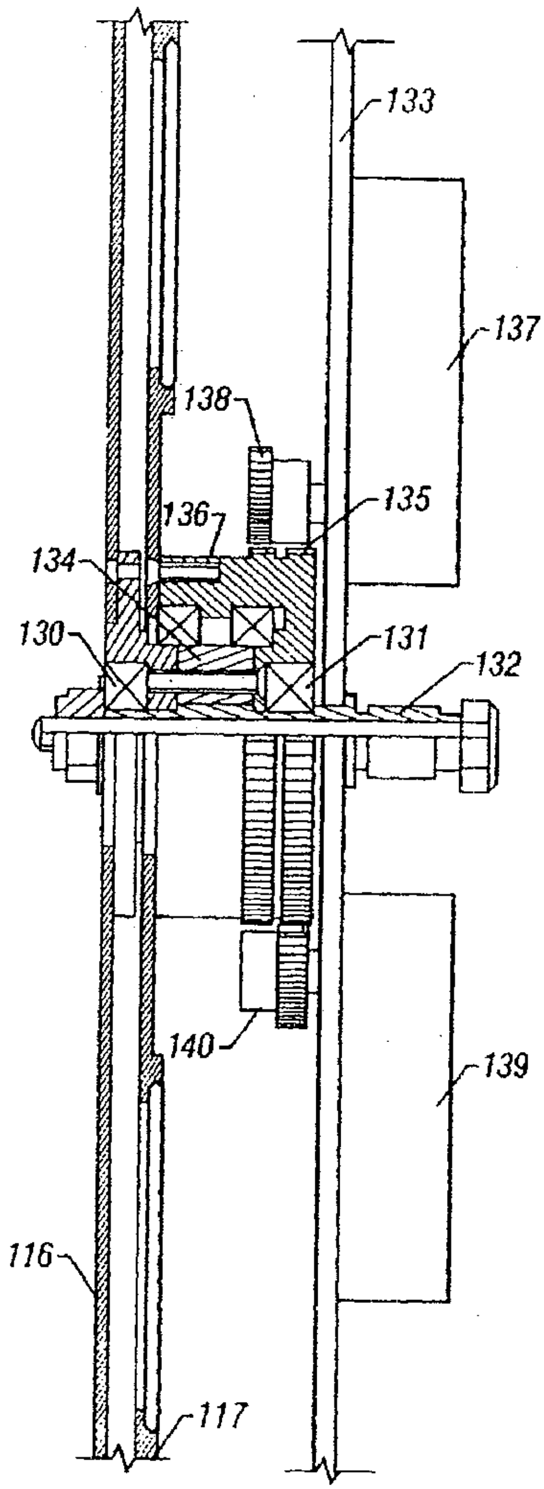


FIG. 12

STAGE LIGHTING LAMP UNIT AND STAGE LIGHTING SYSTEM INCLUDING SUCH UNIT

This application is a continuation (and claims the benefit of priority under 35 USC 120) of U.S. application Ser. No. 10/007,008, filed Dec. 4, 2001 (now U.S. Pat. No. 6,597,132), which is a continuation of U.S. application Ser. No. 09/313,418, filed May 17, 1999 (now U.S. Pat. No. 6,326,741), which is a continuation of U.S. application Ser. No. 08/994,036, filed Dec. 18, 1997 (now U.S. Pat. No. 5,921,659), which is a divisional of U.S. application Ser. No. 08/576,211, filed Dec. 21, 1995 (now U.S. Pat. No. 5,788,365), which is a continuation of U.S. application Ser. No. 08/077,877, filed Jun. 18, 1993 (now U.S. Pat. No. 5,502,627).

This invention relates to stage lighting and is particularly concerned with the control of multiple functions of a lamp.

It has already been proposed to incorporate in a lamp unit a plurality of different functions, such as colour changers, focusing lenses, iris diaphragms, gobo selectors and pan and tilt mechanisms which are controlled from a remote console. Stage lighting systems have as a result reached very high levels of complexity requiring a very complicated main control console and lamp unit constructions. The use of microprocessors, both in the console and the lamps has become conventional as increasing complexity makes it more difficult to produce and subsequently maintain a system which uses hard wired, logic or analog controls. In such systems the microprocessor in the console is used to allow the user to set up lighting cues and to control the sending of appropriate data to the lamp microprocessors. The lamp microprocessors are also involved in controlling communication between the console and the lamps, and also have to control a plurality of servo-motors which drive the various functions of the lamps.

It is one object of the present invention to provide a lamp microprocessor and servo-control arrangement which allows complex functions to be carried out.

It is another object of the invention to provide a lamp control system in which control of pan and tilt movements of each lamp can be carried out in rapid and efficient manner, enabling large groups of lamps to make co-ordinated movements.

It is yet another object of the invention to provide each lamp in a stage lighting system with a means for quickly interrupting its light beam and quickly re-establishing the beam so that a group of lamps can be made, when required to flash in synchronism.

In accordance with one aspect of the invention there is provided a lamp unit for connection to a remote control console for the control of a plurality of different functions of the lamp, said unit comprising a main processor circuit, associated with a communication controller for accepting message data from the console, a plurality of servo-controls for operating said functions of the lamp, and a plurality of co-processors which are-connected to the main processor circuit so as to be supplied thereby with desired value data for the various lamp functions, said servo-controls being controlled by said co-processors.

In the case of pan and tilt controls where close control is required throughout the movement of the lamp from an initial position to a new position, one of the co-processors is assigned solely to the control of movement about each axis. Other functions can share a co-processor.

The main processor circuit of the lamp is preferably programmed to accept data from the control console defining not only a target position for any function, but also a duration over which the function is to be executed. In this case the main processor circuit divides the "journey" into segments and updates the target position data passed to the associated co-processor at intervals.

In accordance with another aspect of the invention, there is provided a lighting control apparatus comprising the combination of a main control console for accepting user input relating to required beam movements, a plurality of independently operable lamp units situated remotely from the console, each of the lamp units incorporating a servo-mechanism for automatically moving the lamp beam about two mutually transverse axes to a desired angular position and data communication means connecting the console to the lamp units for the transmission of desired position data to the lamp units, the desired position data being transmitted in the form of a set of three dimensional linear co-ordinates defining a point in space through which the lamp beam is required to pass, and each lamp unit including a calculating device for calculating the desired angular position from the desired position data and supplying the servo-mechanism with such desired angular position.

In addition to the "point at" mode of operation mentioned above, additional modes may be specified in which the lamps point away from the specified point or in which they all point in the same direction parallel to a line between a fixed position in the co-ordinate system and the specified point.

Conveniently, all the data concerning the positions and orientations of the individual lamp units within the co-ordinate system is stored in a set-up file kept on a hard disk drive in the console. When the same lighting set-up is used at different venues, where it is impossible to set the frame which carries all the lamp units at exactly the same position as that for which the set-up was designed, offset data can be input at the console and either used within the console microcomputer to correct the position data stored during set-up as it is sent out, or such data can be sent to all or the lamp units over the network and stored there, to enable the corrections to be made in the individual lamp processor units.

In accordance with another aspect of the invention, a stage lighting unit comprises a housing, a light source within said housing, an optical system for forming light from said light source into a beam, a rotary shutter device having a plurality of blades, said shutter device being rotatably mounted in the housing so as to cause said blades to pass through and obstruct said beam as the shutter device rotates, a motor for rotating said shutter device and a servo-control for controlling said motor in accordance with data received in use from a remote control console.

The invention also resides in a stage lighting system incorporating a plurality of lighting units as defined above controlled by a common remote control console via data communication means, whereby the rotary shutter devices of all the units can operate in synchronism.

An example of the invention will now be described with reference to the accompanying drawings, in which:

FIG. 1 is a block diagram of a stage lighting system;

FIG. 2 is a block diagram of the internal circuitry of one of a plurality of lamp units in the system of FIG. 1;

FIGS. 3 and 4 are more detailed circuit diagrams showing a pan motor drive control forming part of the internal circuitry of the lamp;

FIGS. 4 to 7 are detailed circuit diagrams showing a rotary shutter motor drive control forming part of the internal circuitry of the lamp;

FIG. 8 is a diagrammatic, part-sectional view of one of the lamps;

FIG. 9 is a perspective view of a pan movement drive arrangement;

FIG. 10 is a perspective view of a tilt movement drive arrangement;

FIG. 11 is a diagrammatic perspective view of the internal moving parts of the lamp;

FIG. 12 is a sectional view showing the drive arrangement for a shutter and a gobo wheel forming part of the lamp; and

FIG. 13 is an elevation of a shutter wheel forming part of the lamp.

Referring firstly to FIG. 1, the system consists basically of a console unit 10; a signal distribution unit 11 and a plurality of lamps L1, L2, L3 . . . , L31, L32, L33 . . . , L61, L62 . . . individually connected by twisted pair data communication links to the distribution unit.

The console unit 10 has an array of switches, slider potentiometers, rotary digital encoders and other user actuable input devices (not shown) and a display indicated at 13. These are all connected to main console cpu 14 (an MC68020 micro-processor) which has the task of receiving inputs from the user actuable input devices and controlling the display. Both tasks are assisted by separate co-processors which directly interface with different parts of the console.

The main cpu can communicate with a hard disk drive unit 15 via a SCSI bus 16 which also connects it to the distribution unit and to an external SCSI port 17, through the intermediary of which the console can, if required be connected to a personal computer. The user controls can be used in setting up a sequence of cues in advance of a performance, the sequence being stored in a cue file on the hard disk drive unit 15. The sequence can be recalled during the performance to enable the various stored cues to be executed. Direct manual control of the lamps from the console is also possible as is manual editing of cues called up from the hard disk. The main console cpu 14 creates messages to be sent to the individual lamps, each message comprising a fixed number of bytes for each lamp. The messages contain data relating to the required lamp orientation, beam coloration, iris diaphragm diameter, gobo selection and rotation, zoom projection lens control and opening or closing of a shutter included in the lamp. A block of the RAM of the main cpu is set aside for the storage of these messages, the block being large enough to contain messages for 240 lamps, being the largest number which can be controlled via the distribution unit. Where it is required to control more than 240 lamps additional distribution units can be connected to the SCSI bus and extra main cpu RAM reserved for message storage. When any message data is changed the main cpu 14 sets a flag in the RAM block which is detected at a given point in the main cpu program loop and interpreted as a signal that the changed message data is to be transferred to the distribution unit 11.

The distribution unit 11 has a main cpu 19 which controls reception of data from the SCSI bus interface and distribution of such data to up to eight blocks of dual,port memory DP1, DP2, DP3 . . . via an eight bit data bus 20. The cpu 19 is alerted to the waiting message data when cpu 14 selects the distribution unit. The cpu 19 then supervises byte by byte transfer of the message data which it routes to the various blocks of dual port memory.

For actually sending out the message data to the lamps, there are a plurality of serial communication controllers SCC1 to SCC30, SCC31 to SCC60 etc, there being thirty serial communication controllers associated with each block of dual port memory. A further cpu DCPU1, DCPU2, etc is associated with each block of dual port memory and distributes message data transferred to the dual port memory to the individual serial communication controllers and the messages are transferred to the lamps. Each serial communication controller in the distribution unit includes a line driver which can be disabled except when data is to be transmitted. Enabling of the driver can cause a spurious signal to be transmitted over the data link. To allow such spurious signals to be identified and ignored, a two-byte gap is left between enabling the line driver and commencing transmission of the message data for the channel in question.

This will be described in more detail herein. All asynchronous serial communication systems require framing information to synchronize the reception process. This has been typically done in the prior art using start bits and stop bits.

The present invention preferably uses FM0 coding in which the data is transmitted as one cycle of the carrier frequency for a zero or as a half cycle of the carrier frequency for a one. When the line has been idle, no waveform at all is present. When the line drivers are first enabled, an arbitrarily short pulse will usually appear on the line, due to lack of synchronization between the data signal and the enabling signal. This short data pulse could be misinterpreted as a start bit, for example and if so it would disturb later framing.

The present invention avoids any problems from this arbitrarily short pulse. To avoid this, the present invention uses a timer on the receive line, set to the time needed to receive two bytes on the serial data line. This timer is restarted whenever a byte on the data line is detected.

Each time the timer interrupt occurs, the number of bytes received is checked against the number of bytes in a valid data frame. If the number is incorrect, then the count is cleared and the message is discarded. If correct, the information is passed to the main program loop by setting a flag variable.

When the data line is first enabled, the distribution box has an internal delay of at least two byte times, which must elapse before any data will be sent. Any data received by the lamp will therefore be discarded as noise by the timer interrupt routine. After that, the real data can be safely sent down the line since the start bit of the first byte will be received correctly. When the transmission is completed, the line drivers will be disabled again.

Each of the cpus eg DCPU1, transfers data from the associated dual port RAM DP1 to the serial communication controller SCC1 to SCC30 with which it is associated one byte at a time, ie the first byte for SCC1 is transferred followed by the first byte for SCC2 and so on, each serial communication controller commencing transmission as soon as it has received its byte of data. The serial communication controllers operate to transmit data at 230.4 Kbps so that it takes about 35 μs to transmit each byte. Transfer of data from the dual port RAM DP1 to the serial communication controllers is, however, at a rate of several Mbps, so that the transmissions from all the serial communication controllers are almost simultaneous. The cpu DCPU1 is not required to monitor the transmission of data by the serial communication controller, but utilizes a software timer to commence transfer of the second byte to the serial communication controllers. This timer is started when transfer of the byte of data to the last serial communication controller SCC30 has been completed and its time-out duration is slightly longer than the byte transmission time, say 40 μs. Transmission of all the messages takes about 1.5 ms out of a distribution unit main program loop duration of 4 ms.

As shown in FIG. 2, each lamp includes a serial communication controller **20** which controls reception of message data from the individual data link connecting it to the distribution unit **11**. The receipt of any signal from the data link causes an interrupt of the lamp main cpu **21** (another MC68000) and the cpu **21** then controls acceptance of the signals. A timer **22** times the gaps between bytes received from the data link and this timer causes another interrupt on time-out. The time-out time of the timer is between the times taken to transmit 1 and 2 bytes, so that time out always occurs following a spurious signal caused by line driver enabling. The time-out interrupt causes the cpu **21** to inspect the total number of bytes received since the initial interrupt and if this is less than the expected number of bytes (which is constant) the message is ignored. The time-out interrupt also resets a software data pointer to the beginning of a receive buffer in readiness for the next transmission.

The cpu **21** operates in accordance with programs stored in the lamp cpu ROM. On receipt of a message of valid length, a program variable representing the number of messages received since the lamp program was last started is incremented and the main program loop of the lamp cpu checks this variable every 6 mS. If the variable has changed since the last check, the data in the receive buffer is compared with corresponding values of variables representing current "desired values" of the various lamp function parameters. For example the receive buffer may contain two bytes representing the x, y and z co-ordinates of a point in an orthogonal three dimensional frame of reference, through which point it is required that the axis of the lamp beam should be directed. If the values of the corresponding byte pairs in the receive buffer and the desired value variables already contained in the cpu RAM are the same, no action is taken in respect of the control of the motors which control pan and tilt action of the lamp (to be described in more detail hereinafter).

As shown in FIG. 2, the main lamp cpu **21** communicates via serial data links **25***a*, **25***b*, **25***c* and **25***d* with four servo-control co-processors **26**, **27**, **28** and **29**. Each of these co-processors is a TMS77C82 cpu. Co-processors **26** and **27** respectively control pan and tilt operation, and each of the co-processors **28** and **29** can control up to six different dc servo-motors operating different functions of the lamp.

Before proceeding with a more detailed description of the circuitry and operation of the lamp electronics, some detail will be given of the various functions of the lamp. FIG. **8** shows the relative positions of a plurality of independently operable beam characteristic control elements within the lamp housing **100**. The lamp housing is pivotally mounted on a U-bracket **101**, which is itself pivotally mounted on a mounting base **102**. FIG. **9** shows the mounting base **102** which incorporates a pan drive motor/gearbox/optical encoder arrangement **104** which drives a gear **105** attached to the U-bracket via a reduction toothed belt drive **106**. FIG. **10** shows how, within the hollow structure of the U-bracket **101**, there is mounted a tilt drive motor/gearbox/optical encoder **107** which drives a gear **108** attached to the lamp housing via another reduction toothed belt drive **109**.

As shown in FIGS. **8** and **11**, within the lamp housing, a light source **110** is mounted within an ellipsoidal reflector **111** providing a light beam with an axis **112** which is reflected by a mirror **113**, which is a dichroic mirror that reflects only visible light and passes ultra voilet and infra red light, the reflected light passing out through an opening **114** at the opposite end of the housing. The reflector **111** has a generally cup-shape surrounding the bulb **110**. According to one aspect of the invention, the axis **112** has an angle pointing in a direction rearward relative to a perpendicular to the central axis **120** of the lamp unit. If the reflector is located as shown, such that an outside edge of the reflector is generally parallel to a rear end of the housing, the optimal packing efficiency is achieved. As shown in FIG. **8**, this allows the reflector to be most efficiently packed into the available space. The reflected beam from the mirror **113** passes firstly through a collimating lens **113***a*, and then the colour changer **115** which comprises dichroic filters having differing transmission characteristics mounted on co-centered three filter disks **115***a*, **115***b* and **115***c* rotable around a common axis of rotation. Each disk has nine different filters on it and one blank space around its periphery, so that up to 1000 different combinations of filters can be positioned across the beam by selective positioning of the three disks (although not all of these combinations are necessarily useful as some may block all visible light). The blank space of each of the disks can be used to eliminate any color changing characteristic of that disk. These disks are driven by three of the dc servo-motors. Next the light beam passes through the plane of a bladed shutter **116** (shown in FIG. **13**) and a first gobo wheel **117** which has various gobos mounted in or over circular holes therein. As shown in FIG. **12** described in more detail hereinafter, two motors are committed to driving the shutter **116** and the gobo wheel **117** respectively. Next, there is a second gobo wheel **118** on which there are mounted a plurality of gobos which are rotatable relative to the wheel **118**. There is one motor (not shown) for driving the gobo wheel **118** and another for rotating the gobos mounted thereon through a gear arrangement (not shown). Next along the light beam is a beam size controlling iris diaphragm **119** driven by another motor (not shown). Two further motors (not shown) drive two lens elements **120**, **121** along guides **122**, **123** parallel to the beam axis using lead screws **124**, **125**. The lens elements form a simple two element zoom lens controlling the spread and focus of the beam. Finally, an outer iris diaphragm **126** is provided adjacent the opening **114** and this is driven by a further motor (not shown). In the example described, therefore only eleven channels are actually employed.

Referring now to FIG. **12**, the shutter **116** is rotatably mounted on bearings **130**, **131** on a shaft **132** fixed to a mounting panel **133** which is secured to the housing. The gobo wheel **117** is rotatably mounted on bearings on a tubular shaft **134** which acts to space the shutter **116** from a first drive gear **135**. The gobo wheel **117** is actually mounted on a second drive gear **136**. The shutter motor **137** (which is combined with a reduction gearbox and an optical encoder) is mounted on the panel **133** and drives a pinion **138** meshed with the first gear **136**. Similarly motor **139** drives a pinion **140** meshed with the second gear **136**. The shutter has four blades arranged symmetrically around its axis, with the blades and the gaps between them each subtending 45 degrees at the axis. The blades and the gaps between them are wide enough to block or clear the entire cross-section of the beam, shown in FIG. **13** at **116***a*.

Turning now to FIGS. **3** and **4**, the co-processor **26** is shown providing an eight bit data output to a d/a converter **40** (FIG. **3**) the output of which is amplified by an operational amplifier **41** and supplied to the "COMPEN" terminal of an LM3524 pulse width modulator ic **42** (FIG. **4**). The ic **42** control a P-channel enhancement mode MOSFET Q1 which, when switched on, connects a 24V supply to a motor supply bus **43** through the intermediary of an inductor **44**. The motor is connected in a bridge formed by two push-pull pairs of MOSFETs Q2, Q3 and Q4, Q5. These four MOSFETs are driven by respective driver transistors Q6, Q7, Q8

and Q9. Transistors Q7 and Q9 are respectively controlled by "LEFT" and "RIGHT" outputs taken from the co-processor **26**, so that FETs Q2 and Q5 or FETs Q3 and Q4 are biassed to conduct. Transistors Q6 and Q8 are driven from a 40V supply rail so as to ensure that FETs Q2 and Q4 are turned hard on when conductive, thereby ensuring minimum power dissipation in these devices.

The two FETs Q3 and Q4 are connected to the return bus via a current sensing resister RC, which supplies a current related signal to a voltage comparator **45** with hysteresis to provide an input to the **A6** input terminal of the co-processor **26** when the current exceeds a predetermined limit. This enables the co-processor to reduce the power applied to the motor to maintain it within safe operating limits.

The optical encoder of the pan motor provides two digital outputs in quadrature, these outputs being cleaned up by interface circuits and applied to two inputs of an HCTL-2016 counter ic **46** intended specifically for use with quadrature type encoders. The counter **46** counts up when the pulses are in one relative phase relationship and down when the opposite phase relationship exists. It therefore maintains a count-state related to the motor shaft position and hence the pan angle of the lamp. This count-state is applied to the **C0** to C7 terminals of the co-processor **26**. The co-processor **26** also receives "desired value" data from the main lamp cpu **21**, via a 75176 ic **47** (which in fact serves both co-processors **26** and **27**). The ic **47** is used to control the transmission of data between the main lamp cpu and the co-processors. Normally the ic **47** is set to receive data from the cpu **21** and pass it to the two co-processors **26** and **27**. At power-up or when the main lamp cpu **21** transmits a "break" command, the co-processor **26** is reset by a circuit **48**. The co-processor **26** has a cycle time of 1 mS and on receipt of new data it determines the distance to be travelled and then increases the "desired position" value which is compared with the actual position count by one sixteenth of the required change on each successive iteration of its control loop.

The desired value signals passed from the cpu **21** to the co-processor **26** are also time-sliced, being incremented every 16 mS. When new position data is transmitted to the lamp it is accompanied by data representing the length of time over which the movement is to be spread. The data is received, as mentioned above, in the form of two byte numbers respectively representing the x, y and z co-ordinates of a point in a Cartesian co-ordinate system. During initial setting up of the system, each lamp is sent data which informs its cpu **21** of its position in the co-ordinate system and also of its orientation.

On receipt of a new set of "point at" co-ordinates, the cpu **21** undertakes a "time-slicing" operation to determine how data should be passed to the co-processors **26** and **27**. First of all, it determines how many 16 mS loops will take place in the time duration determined by the data contained in the massage received by the lamp and sets up a variable U equal to the reciprocal of this number. A travel variable P is initialised to zero and the total distance to be travelled is determined for each of the pan and tilt movements. Thereafter, on every iteration of the 16 mS loop the travel variable P is incremented by the reciprocal variable U, the result is multiplied by the total travel required and this is added to (or subtracted from) the previous desired value before transmission to the co-processor **26** or **27**. When the variable P exceeds unity, the target has been reached.

The message sent to the lamp may include a flag indicating whether travel is to occur in a linear fashion as described above or have a sinusoidal profile imposed on it. In the latter case the value of P is modified as follows:

$P'=\sin(2*P)+0.5*(P>0.5)$ the latter term being 0 or 1

The main cpu **26** must next convert the x,y,z values into pan and tilt value data for passing to the co-processors **26** and **27**. The cpu first carries out a linear transformation of the absolute x,y,z co-ordinates into co-ordinates x',y',z' relative to the lamp's own frame of reference using the data supplied during initial set up. The ratio of the transformed x' and y' values is calculated as a 16-bit integer, which is used as an index to an ARCTAN table stored in ROM to obtain a value for the desired pan angle. To find the tilt angle, it is first necessary to establish the radial position of the target point in the transformed horizontal plane by calculating the square root of the sum of the squares of the co-ordinates x' and y'. In carrying out this calculation it is necessary to detect an overflow condition which exists if the sum of the squares is a 33 bit number. If this condition is detected, each square is divided by four and a new sum is formed, an overflow flag being set to indicate that overflow has occurred. The square root is found by up to sixteen steps of successive approximation and the result is doubled if the overflow flag was set during the calculation. The resulting square root is divided by the value z' and the result is applied as before to the ARCTAN table to determine the tilt angle. The results obtained represent the new pan and tilt positions to which the lamp is to be moved.

The arrangement described for sending out x, y and z co-ordinate data instead of pan and tilt angle data is highly advantageous in that it enables the console main cpu load to be significantly reduced and also makes it very easy for a console operator to control light beam movements. It is frequently required for a group of lamps to be used together to illuminate a single performer. Where the performer moves from one position on stage to another it is required for all the lamps to change position simultaneously to follow. If the system involved transmission of pan and tilt angle data, this data would be different for every lamp in the group. It would have to be set up by the console operator and stored in cue files on the hard disk drive unit **15**. This would be a very time consuming operation as the pan and tilt angles for each lamp would have to be established and recorded individually. The cue record would need to be of considerable size to record all the different data for each lamp. With the arrangement described above, however, only the x,y,z co-ordinate data needs to be stored and when the cue is recalled the same data is sent to each of the lamps in the group.

Whilst it is theoretically possible to use stored cue data in x,y,z co-ordinate form and to use the console main cpu **14** to calculate the pan and tilt angles to send to the lamps, this would be unsatisfactory as the calculations involved would impose a very heavy load on the cpu **14**, particularly where a large number of lamps in several different groups had to be moved as the result of a single cue.

As described above a "point-at" mode is envisaged as the normal operating mode. However, other modes of operation are also envisaged. For example, the lamp could be instructed to point away from the point specified or to point in a direction parallel to a line joining a fixed point (eg the origin of the co-ordinate system) to the point specified. These "point-away" and "point parallel" modes would be selected by means of flags included in the data transmitted to the lamps.

The arrangement described enables the lamps to be very precisely synchronised. The data is transmitted from the distribution unit to all of the lamps simultaneously and each

lamp can start to respond at the end of the message. This enables very precise direction of all the lamps to a moving point in "point-at" mode and very clean parallel sweeps to be made in "point parallel" mode.

It should be noted that the use of x,y,z co-ordinates is also very advantageous in situations where a pre-arranged lighting performance is to be used in several different venues. The pre-loaded gantries or trusses used for such touring performances cannot always be mounted at exactly the required positions relative to the stage because of local conditions. In this case all that is needed is for offsets data to be sent to the lamps at set-up time to enable each lamp cpu to correct its position data. No editing of the individual pre-recorded cues is necessary as it would be in the same circumstances if pan and tilt data were stored.

As part of the set-up procedure for each performance it is necessary to initialise the values of the actual pan and tilt angle count-states, since encoders of the type used do not give any absolute position data. This is accomplished by driving the lamp to an end stop in one direction for each movement. The lamp is driven back to a predetermined number of counts and the counters are reset to zero at this position.

Turning now to FIGS. 5 to 7, the circuitry for controlling the individual dc servo-motors inside the lamp is more complex as each co-processor has to deal with up to six servo-motors. As shown in FIG. 5, the co-processor **28** controls a number of data routers **50** to **54** which determine which channel is being controlled at any given time. The router **50** co-operates with six HCTL-2016 counters **55** which count the quadrature pulse outputs of the respective encoders, to determine which of the counters should supply its count-state to the co-processor **28**. Router **51** controls individual resetting of the counters **55**. Router **52** co-operates with a 74HC175 ic **56** (one for each channel) to determine which L6202 ic motor controller **57** is enabled and also routes "RIGHT" and "LEFT" signals from the co-processor to the circuits **57**. Router **53** controls routing of position error data calculated by the co-processor **28** for each channel to latches **58** (one for each channel) at the input of pulse width modulator circuits for controlling the motor controllers **57**. This error data is actually passed to the latch **58** in an inverted form, so that the larger the error, the smaller the value passed is. Router **54** routes various digital sensor signals to a sensor input of the co-processor, Such sensors are utilized by some of the channels to indicate when the moving part in question is in a datum position. This is required for the gobo wheels, the colour wheels and the shutter, but not for the iris diaphragms or lenses which can be moved to end stop positions. During datum set-up the sensors (optical sensors sensing a hole or flag or Hall effect sensors) are detected and the HCTL counters are reset.

As co-processor **28** has only 256 bytes of internal memory, extra memory is required for each channel to store program variables. The RAM selection control circuit is shown in FIG. 7. The memory ic **59** (an HM6116LP ic) has **11** address lines of which eight are connected to the co-processor write bus via a latch circuit **60** and the remaining three or which are connected to spoare outputs of three of the ics **56**. Spare outputs of the selectors **50, 51, 52** are connected to control terminals of the memory ic and a spare output of the selector **53** is connected to an output enable terminal of the latch circuit **59**. Thus a particular address in the memory ic can be selected by the co-processor by first setting the ics **56** and the selectors **50, 51, 52** to appropriate states and then outputting the lower bytes of the address to latch **60** whilst output from latch **60** is enabled. Two further eight bit latches **61** and **62** provide temporary storage for data to be written to and data just read from the memory ic **59**. When neither reads nor writes are required the memory data bus is tri-stated. Bus contention is thus avoided.

Circuit **57** actually controls the motor current, but it in turn is controlled by a pulse width modulator circuit, comprising the latch **58** and a digital comparator **65** which compares the contents of latch **58** with the count-state of an 8-bit continuously running counter **66***a*, **66***b* serving all channels. The comparator output goes high when the count-state exceeds the latch contents, so that if the latch content is low the comparator output is high for a high proportion of each cycle of the counter **66***a*, **66***b*. The output of the comparator **65** is ANDed with an enable output from ic **56** by a gate **67** and then with the output of an overcurrent detector circuit **68** by another gate **69**.

When a new target value for one of the parameters controlled by co-processor **58** arrives in the receive buffer, and it is associated with execution duration data (this may apply to lens movements, colour changer movements, gobo movements and iris diaphragm movements, but not shutter movements) the cpu **21** handles time slicing as in the pan and tilt operations. Since several channels are controlled by each co-processor, however, no interpolation by the co-processor is used. Instead each channel has its error checked and a new value written (if necessary) to latch **58** every 12 mS

In the case of the shutter, the message received by the lamp merely includes a shutter open or shutter closed command. When the required shutter status changes, the main cpu merely increases the target shutter angle by 45 degrees (in the case of a four bladed shutter) and passes the new value to the co-processor.

This arrangement enables the shutters of some or all of the lamps to be operated in synchronism. Moreover, the console cpu **14**, can operate to update the shutter open/closed instructions at regular intervals to obtain a stroboscopic effect, synchronised for all the lights.

What is claimed is:

1. A method, comprising:

receiving, in a stage lighting lamp which produces light to be transmitted to a target, a value of a parameter from the group of parameters consisting of a movement of an optical element, a movement of a color changer, a movement of a light shaping device, or a movement of an iris diaphragm, which value indicates an amount of movement for the parameter;

dividing said amount of movement into an incremental amount of movement, which will each be carried out in a specified time; and

outputting movement signals at each of a plurality of specified times corresponding to maid amounts of movement determined in said dividing.

2. A method as in claim **1**, wherein each of said incremental amounts of movement corresponds to the same amount of movement.

3. A method as in claim **1**, wherein said dividing and outputting is controlled by a main processor.

4. A method as in claim **1**, wherein said parameter comprises a parameter for multiple lamps, and further comprising outputting movement signals which controls each of said multiple lamps in a synchronized way.

5. A method as in claim **1**, wherein at least one of said incremental amounts of movement is different than another of maid incremental amount of movement.

6. A method as in claim **5**, wherein said amount of movement causes said movement to occur according to a sinusoidal profile.

7. A method am in claim 1, wherein said dividing comprises determining a specified period at which said incremental amount of movement will be updated, determining an incremental amount of movement that is necessary in each specified period, and updating said incremental amount of movement for said each specified period.

8. A method as in claim 7, wherein said specified period is milliseconds.

9. A method as in claim 7, wherein said dividing further comprises maintaining a travel variable representing a total amount of movement, incrementing said travel variable by said amount, and terminating said movement when said travel variable reaches a specified amount.

10. An apparatus comprising:

a lighting unit, which produces an output light beam, and is remotely controllable from a remote console;

a processor, associated with said lighting unit, and operating to receive a control for movement of a lighting control parameter, which is a parameter from the group of parameters consisting of a movement of an optical element, a movement of a color changer, a movement of a light shaping device, or a movement of a diaphragm, and to divide said control into a divided control unit representing an amount of movement for each of said parameters which will occur at each of a plurality of times.

11. An apparatus as in claim 10, wherein said movement of an optical element is a lens movement command.

12. An apparatus as in claim 10, wherein each of said divided control units represents a same amount of movement in each of a plurality of same intervals of time.

13. An apparatus as in claim 10, wherein said processor receives said control and produces a plurality of outputs which control a plurality of units to carry out said movement in a synchronized way.

14. An apparatus as in claim 10, further comprising a controller, producing a plurality of outputs for a plurality of different units representing synchronized control of said different units, each unit including a processor, which controls according to at least one of said plurality of outputs.

15. An apparatus as in claim 10, wherein at least one of said divided control units represents a different amount of movement than for another of said divided control units.

16. An apparatus as in claim 15, wherein said amounts of movement causes said movement to occur according to a sinusoidal profile.

17. An apparatus as in claim 10, wherein said processor divides the movement into a plurality of update periods, each update period being multiple milliseconds, and produces output signals at each of said update periods which cause a controlled unit to move.

18. An apparatus as in claim 17, wherein said processor maintains a travel parameter, which represents a total amount of travel which has occurred so far.

19. A lighting control method, comprising:

receiving a control which commands movement of a parameter associated with projection of light,

determining a time over which said control should be carried out and dividing said control into a plurality of divided units, each of said units representing an amount of movement which should be carried out in a specified divided time, and

outputting signals at each of said specified divided times commanding movement at each of said specified divided times.

20. A method as in claim 19, further comprising maintaining a variable which indicates a total amount of movement at a current time.

21. A method as in claim 19, wherein each of said divided units represents a same amount of movement.

22. A method as in claim 19, wherein at least one of said divided units represents a different amount of movement than another of said divided units.

23. A method as in claim 19, wherein said parameter is a parameter commanding movement of a light shape alteration.

24. A method as in claim 19, wherein said parameter is a parameter commanding changing of a color of light which is projected.

25. A method as in claim 19, wherein said parameter is a parameter commanding changing of a size of a diaphragm.

26. A method as in claim 19, wherein said parameter is a parameter which changes of focusing of light by a lens.

27. A method as in claim 19, wherein said output signals are output to a plurality of different lights and are substantially synchronized with one another.

28. A method as in claim 19, wherein each of said specific divided times occur multiple milliseconds apart.

29. A method as in claim 19, wherein said parameter is a parameter that changes an amount of movement by said light.

30. A system, comprising:

a processor that receives a control that commands movement of a parameter associated with projection of light, and a control that indicates a time for said movement, said processor operating to produce a plurality of divided movement amounts, each of said movement amounts representing a movement which should be carried out at a specific divided time; and

an output signal terminal, which provides an output signal that controls said movement at said divided times.

31. A system as in claim 30, wherein said processor produces each of said divided units which represents the same amount of movement.

32. A System as in claim 30, wherein said processor produces at least one of said divided units representing a different amount of movement than another of said divided units.

33. A system as in claim 30, wherein said control represents movement of a light shape alteration.

34. A system as in claim 30, wherein said control represents movement of a gobo.

35. A system am in claim 30, wherein said parameter represents movement of a color changing device.

36. A system as in claim 30 wherein said parameter represents changing a size of a diaphragm.

37. A system as in claim 30, wherein said parameter represents movement of a lens.

* * * * *

EXHIBIT R

US006934071B2

(12) **United States Patent**
Hunt

(10) **Patent No.: US 6,934,071 B2**
(45) **Date of Patent: Aug. 23, 2005**

(54) **PIXEL BASED GOBO RECORD CONTROL FORMAT**

(75) Inventor: **Mark Hunt**, Birmingham (GB)

(73) Assignee: **Production Resource Group, L.L.P.**, New Windsor, NY (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **10/271,521**

(22) Filed: **Oct. 15, 2002**

(65) **Prior Publication Data**

US 2003/0107795 A1 Jun. 12, 2003

Related U.S. Application Data

(63) Continuation of application No. 09/882,755, filed on Jun. 15, 2001, now Pat. No. 6,466,357, which is a continuation of application No. 09/500,393, filed on Feb. 8, 2000, now Pat. No. 6,256,136, which is a continuation of application No. 09/145,314, filed on Aug. 31, 1998, now Pat. No. 6,057,958.

(60) Provisional application No. 60/059,161, filed on Sep. 17, 1997, and provisional application No. 60/065,133, filed on Nov. 12, 1997.

(51) **Int. Cl.**[7] **G02B 26/00**; G09G 5/00; G06T 13/00; G06K 9/00; G06K 9/46

(52) **U.S. Cl.** **359/291**; 359/290; 345/611; 345/473; 345/604; 345/431; 382/167; 382/181; 382/192; 353/122; 362/282

(58) **Field of Search** 382/203, 154, 382/254, 300, 167, 192, 181, 242; 353/25, 122; 358/532, 518; 362/282, 294; 359/290, 291; 345/611, 619, 419, 431, 473, 604, 610

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,891,738	A	*	1/1990	Richardson et al.	362/282
5,113,332	A	*	5/1992	Richardson	362/282
5,282,121	A	*	1/1994	Bornhorst et al.	362/294
5,828,485	A	*	10/1998	Hewlett	359/291
5,852,443	A	*	12/1998	Kenworthy	345/431
5,953,151	A	*	9/1999	Hewlett	359/291
6,256,136	B1	*	7/2001	Hunt	359/291
6,486,880	B2	*	11/2002	Clark-Schreyer et al.	345/473
6,771,411	B2	*	8/2004	Hewlett	359/298
2003/0147117	A1	*	8/2003	Hewlett	359/292

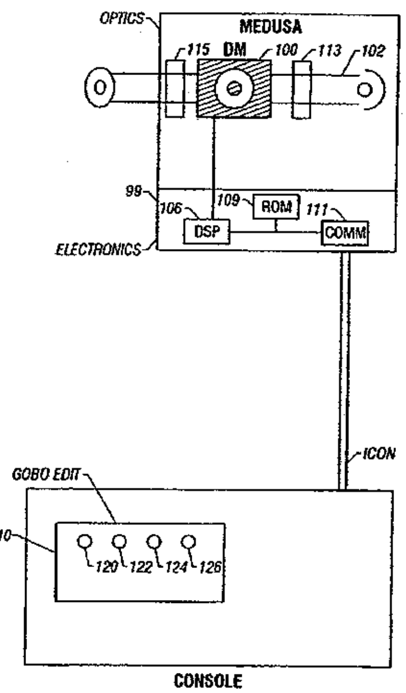
* cited by examiner

Primary Examiner—Loha Ben
(74) *Attorney, Agent, or Firm*—Fish & Richardson P.C.

(57) **ABSTRACT**

A special record format used for commanding light pattern shapes and addressable light pattern shape generator. The command format includes a first part which commands a specified gobo and second parts which command the characteristics of that gobo. The gobo is formed by making a default gobo based on the type and modifying that default gobo to fit the characteristics.

58 Claims, 6 Drawing Sheets



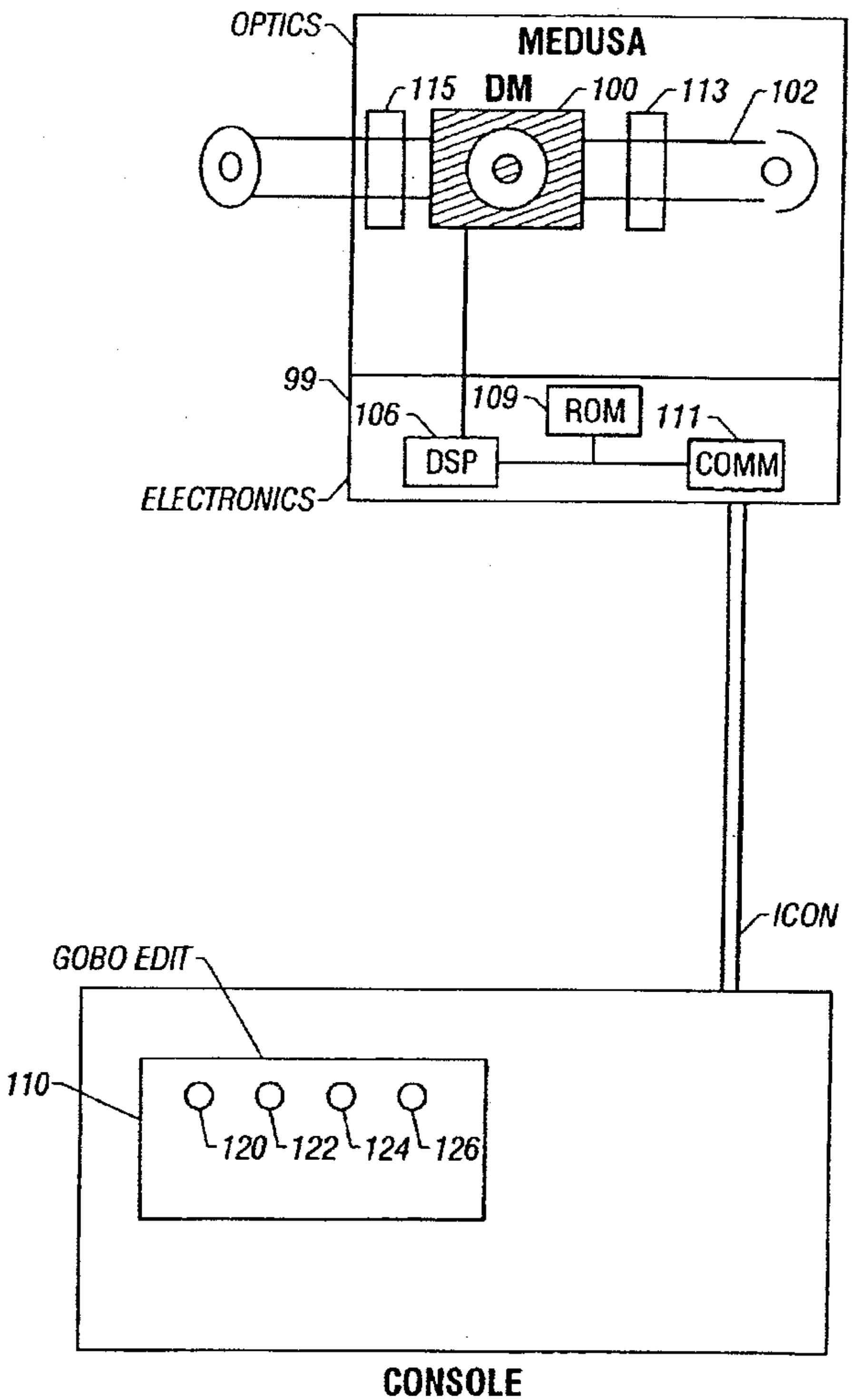

OPTICS
MEDUSA
115
DM
100
113
102
99
106
109
ROM
111
DSP
COMM
ELECTRONICS
ICON
GOBO EDIT
110
120
122
124
126
CONSOLE

FIG. 1

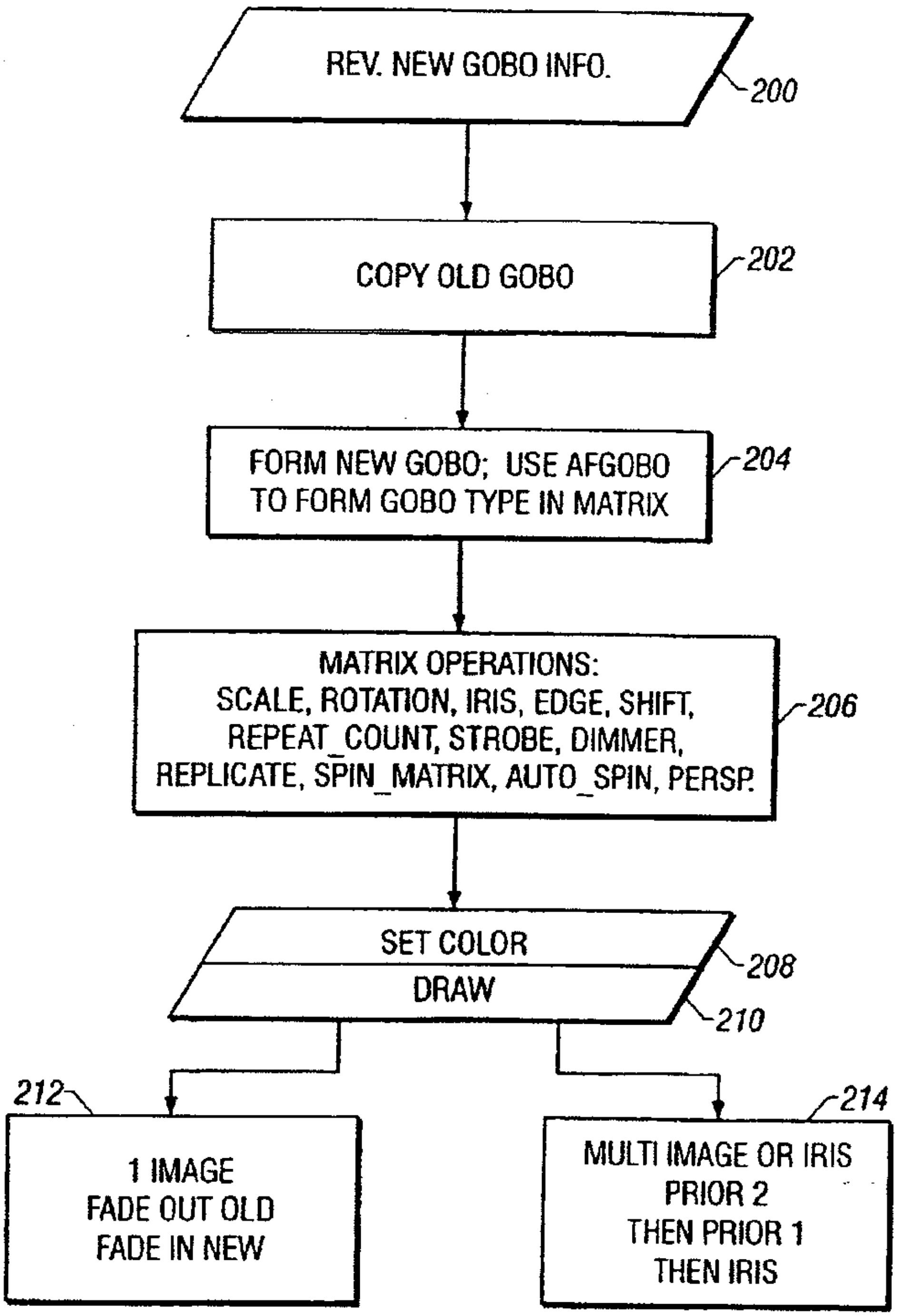

REV. NEW GOBO INFO.
200
COPY OLD GOBO
202
FORM NEW GOBO; USE AFGOBO TO FORM GOBO TYPE IN MATRIX
204
MATRIX OPERATIONS: SCALE, ROTATION, IRIS, EDGE, SHIFT, REPEAT_COUNT, STROBE, DIMMER, REPLICATE, SPIN_MATRIX, AUTO_SPIN, PERSP.
206
SET COLOR
208
DRAW
210
212
1 IMAGE FADE OUT OLD FADE IN NEW
214
MULTI IMAGE OR IRIS PRIOR 2 THEN PRIOR 1 THEN IRIS

FIG. 2

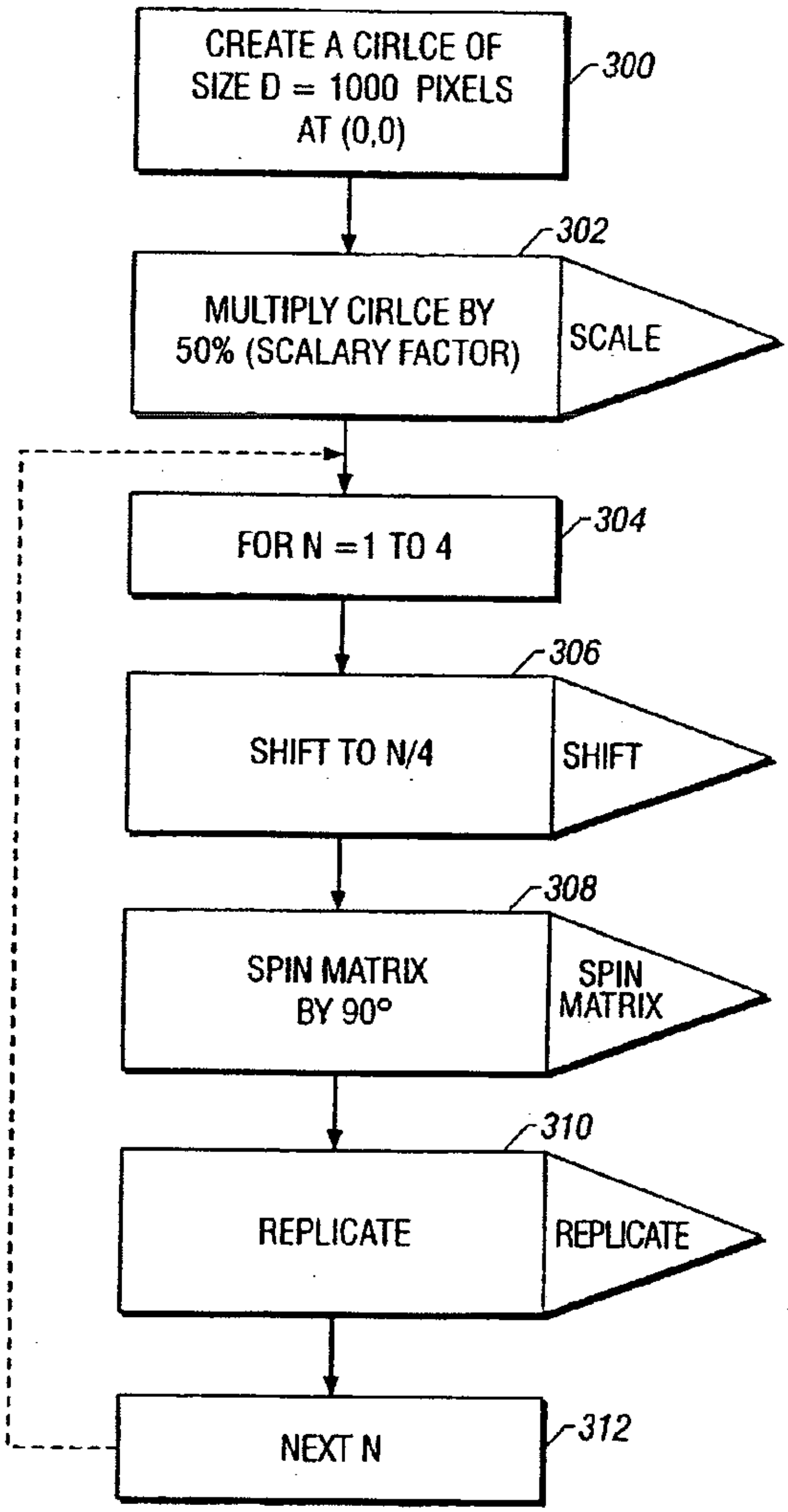

CREATE A CIRLCE OF SIZE D = 1000 PIXELS AT (0,0)
300
302
MULTIPLY CIRLCE BY 50% (SCALARY FACTOR)
SCALE
FOR N =1 TO 4
304
306
SHIFT TO N/4
SHIFT
308
SPIN MATRIX BY 90°
SPIN MATRIX
310
REPLICATE
REPLICATE
NEXT N
312

FIG. 3

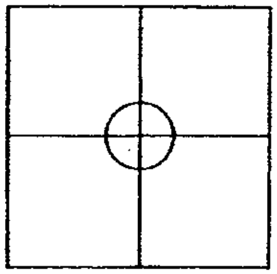


FIG. 4A

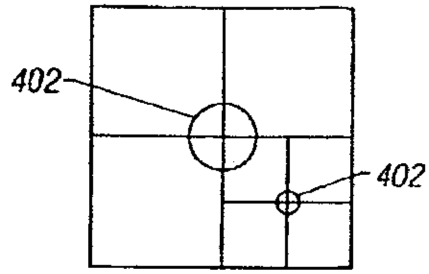


FIG. 4E

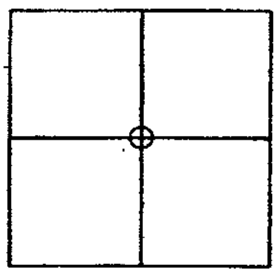


FIG. 4B

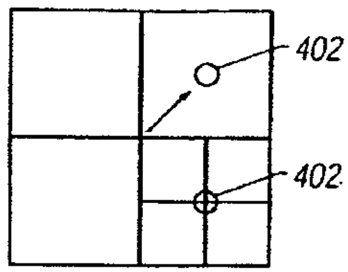


FIG. 4F

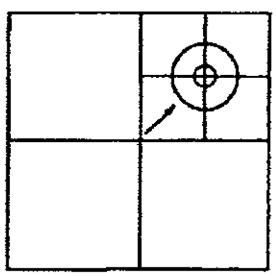


FIG. 4C

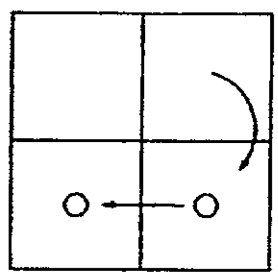


FIG. 4G

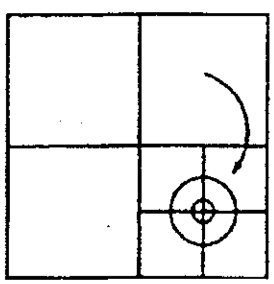


FIG. 4D

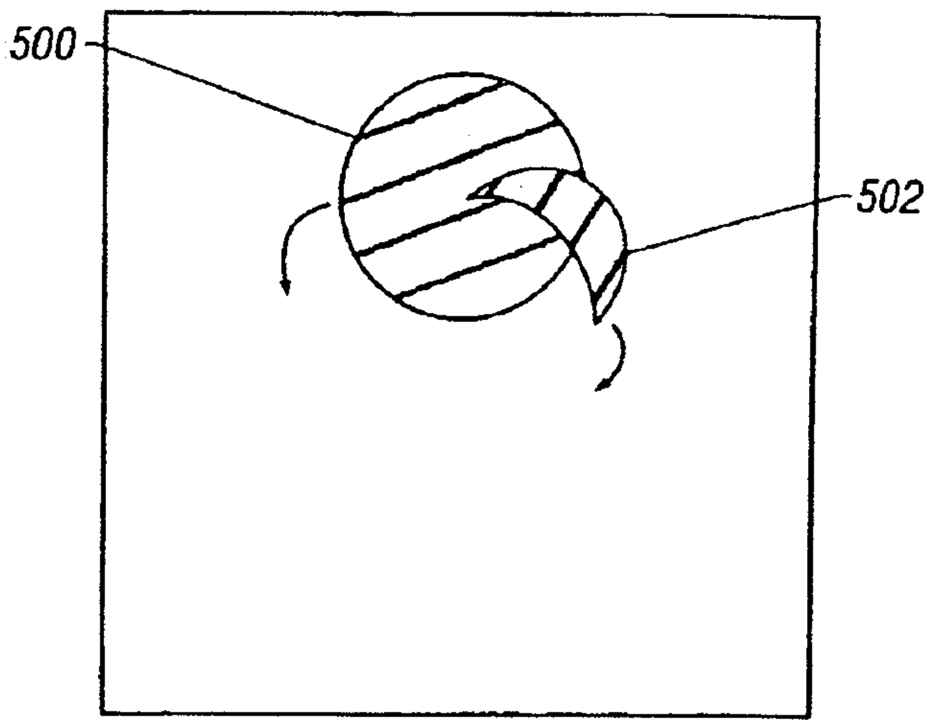


FIG. 5

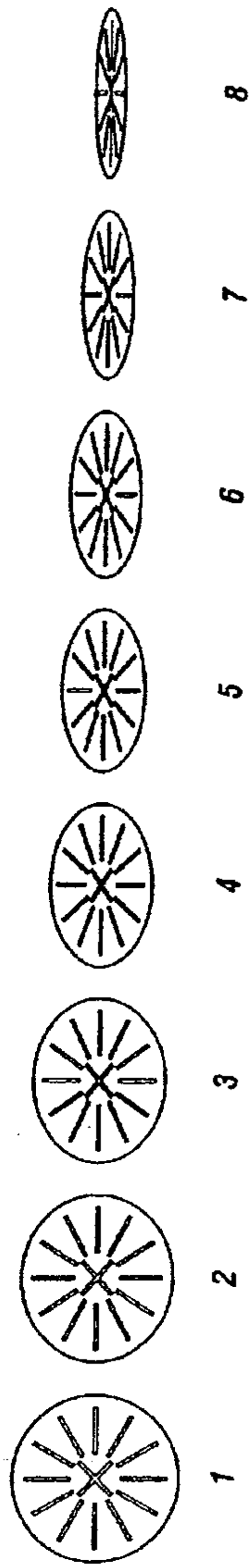


FIG. 6

PIXEL BASED GOBO RECORD CONTROL FORMAT

This application is a continuation of U.S. application Ser. No. 09/882,755, filed Jun. 15, 2001 now U.S. Pat No. 6,466,357, which is a continuation of U.S. application Ser. No. 09/500,393, filed Feb. 8, 2000 now U.S. Pat No. 6,256,136, which is a continuation of U.S. application Ser. No. 09/145,314, filed Aug. 31, 1998 now U.S. Pat No. 6,057,958, which claims priority from U.S. Provisional application Nos. 60/059,161, filed Sep. 17, 1997, and 60/065,133, filed Nov. 12, 1997.

FIELD

The present invention relates to a system of controlling is light beam pattern ("gobo") shape in a pixilated gobo control system.

BACKGROUND

Commonly assigned patent application Ser. No. 08/854,353, the disclosure of which is herewith incorporated by reference, describes a stage lighting system which operates based on computer-provided commands to form special effects. One of those effects is control of the shape of a light pattern that is transmitted by the device. This control is carried out on a pixel-by-pixel basis, hence referred to in this specification as pixilated. Control is also carried out using an x-y controllable device. The preferred embodiment describes using a digital mirror device, but other x-y controllable devices such as a grating light valve, are also contemplated.

The computer controlled system includes a digital signal processor **106** which is used to create an image command. That image command controls the pixels of the x-y controllable device to shape the light that it is output from the device.

The system described in the above-referenced application allows unparalleled flexibility in selection of gobo shapes and movement. This opens an entirely new science of controlling gobos. The present inventors found that, unexpectedly, even more flexibility is obtained by a special control language for controlling those movements.

SUMMARY

The present disclosure defines a way of communicating with an x-y controllable device to form special electronic light pattern shapes. More specifically, the present application describes using a control language to communicate with an electronic gobo in order to reposition part or all of the image that is shaping the light.

BRIEF DESCRIPTION OF THE DRAWINGS

These and other aspects of the invention will now be described with reference to the attached drawings, in which:

FIG. **1** shows a block diagram of the basic system operating the embodiment;

FIG. **2** shows a basic flowchart of operation;

FIG. **3** shows a flowchart of forming a replicating circles type gobo;

FIGS. **4A** through **4G** show respective interim results of carrying out the replicating circles operation;

FIG. **5** shows the result of two overlapping gobos rotating in opposite directions; and

FIGS. **6(1)** through **6(8)** show a z-axis flipping gobo.

DESCRIPTION OF THE PREFERRED EMBODIMENT

FIG. **1** shows a block diagram of the hardware used according to the preferred embodiment. As described above, this system uses a digital mirror device **100**, which has also been called a digital mirror device ("DMD") and a digital light processor device ("DLP"). More generally, any system which allows controlling shape of light on a pixel basis, including a grating light valve, could be used as the light shaper. This light shaper forms the shape of light which is transmitted. FIG. 1 shows the light being transmitted as **102**, and shows the transmitted light. The information for the digital mirror **100** is calculated by a digital signal processor **106**. Information is calculated based on local information stored in the lamp, e.g., in ROM **109**, and also in information which is received from the console **104** over the communication link.

The operation is commanded according to a format.

The preferred data format provides 4 bytes for each of color and gobo control information.

The most significant byte of gobo control data, ("dfGobo") indicates the gobo type. Many different gobo types are possible. Once a type is defined, the gobo formed from that type is represented by a number. That type can be edited using a special gobo editor described herein. The gobo editor allows the information to be modified in new ways, and forms new kinds of images and effects.

The images which are used to form the gobos may have variable and/or moving parts. The operator can control certain aspects of these parts from the console via the gobo control information. The type of gobo controls the gobo editor to allow certain parameters to be edited.

The examples given below are only exemplary of the types of gobo shapes that can be controlled, and the controls that are possible when using those gobo shapes. Of course, other controls of other shapes are possible and predictable based on this disclosure.

A first embodiment is the control of an annulus, or "ring" gobo. The DMD **100** in FIG. **1** is shown with the ring gobo being formed on the DMD. The ring gobo is type 000A. When the gobo type 0A is enabled, the gobo editor **110** on the console **104** is enabled and the existing gobo encoders **120**, **122**, **124**, and **126** are used. The gobo editor **110** provides the operator with specialized control over the internal and the external diameters of the annulus, using separate controls in the gobo editor.

The gobo editor and control system also provides other capabilities, including the capability of timed moves between different edited parameters. For example, the ring forming the gobo could be controlled to be thicker. The operation could then effect a timed move between these "preset" ring thicknesses. Control like this cannot even be attempted with conventional fixtures.

Another embodiment is a composite gobo with moving parts. These parts can move through any paths that are programmed in the gobo data itself. This is done in response to the variant fields in the gobo control record, again with timing. Multiple parts can be linked to a single control allowing almost unlimited effects.

Another embodiment of this system adapts the effect for an "eye" gobo, where the pupil of the eye changes its position (look left, look right) in response to the control.

Yet another example is a Polygon record which can be used for forming a triangle or some other polygonal shape.

The control can be likened to the slider control under a QuickTime movie window, which allows you to manually move to any point in the movie. However, our controls need not be restricted to timelines.

Even though such moving parts are used, scaling and rotation on the gobo is also possible.

The following type assignments are contemplated:

00-0F=FixedGobo (with no "moving parts")

10-1F=SingleCntrl (with 1 "moving part")

20-2F=DoubleCntrl (with 2 "moving parts")

30-FF=undefined, reserved.

The remaining control record bytes for each type are defined as follows:

Byte: memory	dfGobo2	dfGobo3	dfGobo4	#gobos/type,	total
- FixedGobo:	ID [23:16]	ID [15:8]	ID [7:0]	16M/type	256M
SingleCntrl:	ID [15:8]	ID [7:0]	control#1	64k/type	1M
DoubleCntrl:	ID [7:0]	control#2	control#1	256/type	4k

As can be seen from this example, this use of the control record to carry control values does restrict the number of gobos which can be defined of that type, especially for the 2-control type.

Console Support:

The use of variant part gobos requires no modifications to existing console software for the ICON (7M) console. The Gobo editor in current ICON software already provides 4 separate encoders for each gobo. These translate directly to the values of the 4 bytes sent in the communications data packet as follows:

Byte: dfGobo	dfGobo2	dtGobo3	dfGobo4
Enc: TopRight	MidRight	BotRight	BotLeft
FixedGobo:	ID [23:16]	ID [15:8]	ID [7:0]
SingleCntrl:	ID [15:8]	ID [7 0]	control#1
DoubleCntrl:	ID [7:0]	control#2	control#1

These values would be part of a preset gobo, which could be copied as the starting point.

Once these values are set, the third and fourth channels automatically become the inner/outer radius controls. Using two radii allows the annulus to be turned "inside out".

Each control channel's data always has the same meaning within the console. The console treats these values as simply numbers that are passed on. The meanings of those numbers, as interpreted by the lamps change according to the value in dfGobo.

The lamp will always receives all 4 bytes of the gobo data in the same packet. Therefore, a "DoubleCntrl" gobo will always have the correct control values packed along with it.

Hence, the console needs no real modification. If a "soft" console is used, then name reassignments and/or key reassignments may be desirable.

Timing:

For each data packet, there is an associated "Time" for gobo response. This is conventionally taken as the time allotted to place the new gobo in the light gate. This delay has been caused by motor timing. In this system, variant gobo, the control is more dynamically used. If the non-variant parts of the gobo remain the same, then it is still the same gobo, only with control changes. Then, the time value is interpreted as the time allowed for the control change.

Since different gobo presets (in the console) can reference the same gobo, but with different control settings, this allows easily programmed timed moves between different annuli, etc.

Internal Workings:

When the gobo command data is extracted from the packet at the lamp, the dfGobo byte is inspected first, to see

if either dfGobo3 or dfGobo4 are significant in selecting the image. In the case of the "Cntrl" variants, one or both of these bytes is masked out, and the resulting 32-bit number is used to search for a matching gobo image (by Gobo_1D) in the library stored in the lamp's ROM **109**.

If a matching image is found, and the image is not already in use, then the following steps are taken:

1) The image data is copied into RAM, so that its fields may be modified by the control values. This step will be skipped if the image is currently active.

2) The initial control values are then recovered from the data packet, and used to modify certain fields of the image data, according to the control records.

3) The image is drawn on the display device, using the newly-modified fields in the image data.

If the image is already in use, then the RAM copy is not altered. Instead, a time-sliced task is set up to slew from the existing control values to those in the new data packet, in a time determined by the new data packet.

At each vertical retrace of the display, new control values are computed, and steps 2 (using the new control values) and 3 above are repeated, so that the image appears modified with time.

The image data records:

All images stored in the lamp are in a variant record format:

Header:

Length 32 bits, offset to next gobo in list.

Gobo _1D 32 bits, serial number of gobo.

Gobo records:

Length	32 bits, offset to next record.
Opcode	16 bits, type of object to be drawn.
Data	Variant part-data describing object.
_Length	32 bits, offset to next record.
Opcode	16 bits, type of object to be drawn.
Data	Variant part-data describing object.
_ EndMarker	64 bits, all zeroes-indicates end of gobo data.
+ Next gobo, or End Marker, indicating end of gobo list.	

Gobos with controls are exactly the same, except that they contain control records, which describe how the control values are to affect the gobo data. Each control record contains the usual length and Opcode fields, and a field containing the control number (1 or 2).

These are followed by a list of "field modification" records. Each record contains information about the offset (from the start of the gobo data) of the field, the size (8, 16 or 32 bits) of the field, and how its value depends on the control value.

Length	32 bits, offset to next record
Opcode	16 bits = control_record (constant)
CntrlNum	16 bits = 1 or 2 (control number)
	/* field modification record #1 */
Address	16 bits, offset from start of gobo to affected field.
Flags	16 bits, information about field (size, signed, etc)
Scale use	16 bits, scale factor applied to control before
ZPoint	16 bits, added to control value after scaling.
	/* field modification record #2 */
Address	16 bits, offset from start of gobo to affected field.
Flags etc)	16 bits, information about field (size, signed,
Scale use	16 bits, scale factor applied to control before
ZPoint	16 bits, added to control value after scaling.

As can be seen, a single control can have almost unlimited effects on the gobo, since ANY values in the data can be modified in any way, and the number of field modification records is almost unlimited.

Note that since the control records are part of the gobo data itself, they can have intimate knowledge of the gobo structure. This makes the hard-coding of field offsets acceptable.

In cases where the power offered by this simple structure is not sufficient, a control record could be defined which contains code to be executed by the processor. This code would be passed parameters, such as the address of the gobo data, and the value of the control being adjusted.

Example Records.

The Annulus record has the following format:

Length	32 bits
Opcode	16 bits, = type_annulus
Pad	16 bits, unused
Centre_x	16 bits, x coordinate of centre
Centre_y	16 bits, y coordinate of centre
OuterRad	16 bits, outside radius (the radii get swapped when drawn if their values are in the wrong order)
InnerRad	16 bits, inside radius

It can be seen from this that it is easy to "target" one of the radius parameters from a control record. Use of two control records, each with one of the radii as a target, would provide full control over the annulus shape.

Note that if the centre point coordinates are modified, the annulus will move around the display area, independent of any other drawing elements in the same gobo's data.

The Polygon record for a triangle has this format:

Length	32 bits
Opcode	16 bits, = type_polygon
Pad	16 bits, vertex count = 3
Centre_x	16 bits, x coordinate of vertex
Centre_y	16 bits, y coordinate of vertex
Centre_x	16 bits, x coordinate of vertex
Centre_y	16 bits, y coordinate of vertex
Centre_x	16 bits, x coordinate of vertex
Centre_y	16 bits, y coordinate of vertex

It is easy to modify any of the vertex coordinates, producing distortion of the triangle.

The gobo data can contain commands to modify the drawing environment, by rotation, scaling, offset, and color control, the power of the control records is limitless.

Second Embodiment

This second embodiment provides further detail about implementation once the gobo information is received.

Gobo information is, at times, being continuously calculated by DSP **106**. The flowchart of FIG. 2 shows the handling operation that is carried out when new gobo information is received.

At step **200**, the system receives new gobo information. In the preferred embodiment, this is done by using a communications device **111** in the lamp **99**. The communications device is a mailbox which indicates when new mail is received. Hence, the new gobo information is received at step **200** by determining that new mail has been received.

At step **202**, the system copies the old gobo and switches pointers. The operation continues using the old gobo until the draw routine is called later on.

At step **204**, the new information is used to form a new gobo. The system uses a defined gobo ("dfGobo") as discussed previously which has a defined matrix. The type dfGobo is used to read the contents from the memory **109** and thereby form a default image. That default image is formed in a matrix. For example, in the case of an annulus, a default size annulus can be formed at position 0,0 in the matrix. An example of forming filled balls is provided herein.

Step **206** represents calls to subroutines. The default gobo is in the matrix, but the power of this system is its ability to very easily change the characteristics of that default gobo. In this embodiment, the characteristics are changed by changing the characteristics of the matrix and hence, shifting that default gobo in different ways. The matrix operations, which are described in further detail herein, include scaling the gobo, rotation, iris, edge, strobe, and dimmer. Other matrix operations are possible. Each of these matrix operations takes the default gobo, and does something to it.

For example, scale changes the size of the default gobo rotation rotates the default gobo by a certain amount.

Iris simulates an iris operation by choosing an area of interest, typically circular, and erasing everything outside that area of interest. This is very easily done in the matrix, since it simply defines a portion in the matrix where all black is written.

Edge effects carry out certain effects on the edge such as softening the edge. This determines a predetermined thickness, which is translated to a predetermined number of pixels, and carries out a predetermined operation on the number of pixels. For example, for a 50% edge softening, every other pixel can be turned off. The strobe is in effect that allows all pixels to be turned on and off at a predetermined frequency, i.e., 3 to 10 times a second. The dimmer allows the image to be made dimmer by turning off some of the pixels at predetermined times.

The replicate command forms another default gobo, to allow two different gobos to be handled by the same record. This will be shown with reference to the exemplary third embodiment showing balls. Each of those gobos is then handled as the same unit and the entirety of the gobos can be, for example, rotated. The result of step **206** and all of these subroutines that are called is that the matrix includes information about the bits to be mapped to the digital mirror **100**.

At step **208**, the system then obtains the color of the gobos from the control record discussed previously. This gobo color is used to set the appropriate color changing circuitry **113** and **115** in the lamp **99**. Note that the color changing circuitry is shown both before and after the digital mirror **100**. It should be understood that either of those color changing circuits could be used by itself.

At step **210**, the system calls the draw routine in which the matrix is mapped to the digital mirror. This is done in different ways depending on the number of images being used. Step **212** shows the draw routine for a single image being used as the gobo. In that case, the old gobo, now copied as shown in step **202**, is faded out while the new gobo newly calculated is faded in. Pointers are again changed so that the system points to the new gobo. Hence, this has the effect of automatically fading out the old gobo and fading in the new gobo.

Step **214** schematically shows the draw routine for a system with multiple images for an iris. In that system, one of the gobos is given priority over the other. If one is brighter than the other, then that one is automatically given priority.

The one with priority 2, the lower priority one, is written first. Then the higher priority gobo is written. Finally, the iris is written which is essentially drawing black around the edges of the screen defined by the iris. Note that unlike a conventional iris, this iris can take on many different shapes. The iris can take on not just a circular shape, but also an elliptical shape, a rectangular shape, or a polygonal shape. In addition, the iris can rotate when it is non-circular so that for the example of a square iris, the edges of the square can actually rotate.

Returning to step **206**, in the case of a replicate, there are multiple gobos in the matrix. This allows the option of spinning the entire matrix, shown as spin matrix.

An example will now be described with reference to the case of repeating circles. At step **200**, the new gobo information is received indicating a circle. This is followed by the other steps of **202** where the old gobo is copied, and **204** where the new gobo is formed. The specific operation forms a new gobo at step **300** by creating a circle of size diameter equals 1000 pixels at origin 00. This default circle is automatically created. FIG. **4**A shows the default gobo which is created, a default size circle at 00. It is assumed for purposes of this operation that all of the circles will be the same size.

At step **302**, the circle is scaled by multiplying the entire circle by an appropriate scaling factor. Here, for simplicity, we are assuming a scaling factor of 50% to create a smaller circle. The result is shown in FIG. **4**B. A gobo half the size of the gobo of FIG. **4**A is still at the origin. This is actually the scale of the subroutine as shown in the right portion of step **302**. Next, since there will be four repeated gobos in this example, a four-loop is formed to form each of the gobos at step **304**. Each of the gobos is shifted in position by calling the matrix operator shift. In this example, the gobo is shifted to a quadrant to the upper right of the origin. This position is referred to as π over 4 in the FIG. 3 flowchart and results in the gobo being shifted to the center portion of the top right quadrant as shown in FIG. **4**C. This is again easily accomplished within the matrix by moving the appropriate values. At step **308**, the matrix is spun by 90 degrees in order to put the gobo in the next quadrant as shown in FIG. **4**D in preparation for the new gobo being formed into the same quadrant. Now the system is ready for the next gobo, thereby calling the replicate command which quite easily creates another default gobo circle and scales it. The four-loop is then continued at step **312**.

The replicate process is shown in FIG. **4**E where a new gobo **402** is formed in addition to the existing gobo **400**. The system then passes again through the four-loop, with the results being shown in the following figures. In FIG. **4**F, the new gobo **402** is again moved to the upper right quadrant (step **306**). In FIG. **4**G, the matrix is again rotated to leave room for a new gobo in the upper right quadrant. This continues until the end of the four-loop. Hence, this allows each of the gobos to be formed.

Since all of this is done in matrix operation, it is easily programmable into the digital signal processor. While the above has given the example of a circle, it should be understood that this scaling and moving operation can be carried out for anything. The polygons, circles, annulus, and everything else is easily scaled.

The same operation can be carried out with the multiple parameter gobos. For example, for the case of a ring, the variable takes the form annulus (inner R, outer R, x and y). This defines the annulus and turns of the inner radius, the outer radius, and x and y offsets from the origin. Again, as shown in step 3, the annulus is first written into the matrix as a default size, and then appropriately scaled and shifted. In terms of the previously described control, the ring gobo has two controls: control **1** and control **2** defined the inner and outer radius.

Each of these operations is also automatically carried out by the command repeat count which allows easily forming the multiple position gobo of FIGS. **4**A–**4**G. The variable auto spin defines a continuous spin operation. The spin operation commands the digital signal processor to continuously spin the entire matrix by a certain amount each time.

One particularly interesting feature available from the digital mirror device is the ability to use multiple gobos which can operate totally separately from one another raises the ability to have different gobos spinning in different directions. When the gobos overlap, the processor can also calculate relative brightness of the two gobos. In addition, one gobo can be brighter than the other. This raises the possibility of a system such as shown in FIG. **5**. Two gobos are shown spinning in opposite directions: the circle gobo **500** is spinning the counterclockwise direction, while the half moon gobo **502** is spinning in the clockwise direction. At the overlap, the half moon gobo which is brighter than the circle gobo, is visible over the circle gobo. Such effects were simply not possible with previous systems. Any matrix operation is possible, and only a few of those matrix operations have been described herein.

A final matrix operation to be described is the perspective transformation. This defines rotation of the gobo in the Z axis and hence allows adding depth and perspective to the gobo. For each gobo for which rotation is desired, a calculation is preferably made in advance as to what the gobo will look like during the Z axis transformation. For example, when the gobo is flipping in the Z axis, the top goes back and looks smaller while the front comes forward and looks larger. FIGS. **6(1)**–**6(8)** show the varying stages of the gobo flipping. In FIG. **6(8)**, the gobo has its edge toward the user. This is shown in FIG. **6(8)** as a very thin line, e.g., three pixels wide, although the gobo could be zero thickness at this point. Automatic algorithms are available for such Z axis transformation, or alternatively a specific Z axis transformation can be drawn and digitized automatically to enable a custom look.

Although only a few embodiments have been described in detail above, other embodiments are contemplated by the inventor and are intended to be encompassed within the following claims. In addition, other modifications are contemplated and are also intended to be covered.

What is claimed is:

1. A computer system comprising:

a computer based controller producing an output signal that includes parameters that digitally control a shape of a projected light.

2. A system as in claim **1**, wherein said computer based controller includes a memory storing associated information about shapes for said output signal.

3. A system as in claim **1**, wherein said output signal is in a specified format, where different parts of the signal have different information to control said shape.

4. A system as in claim **3**, wherein said format includes a part which controls a specific color of said projected light.

5. A computer system comprising:

a computer based controller producing an output signal that includes parameters that digitally control a shape of a projected light, wherein said output signal includes a first part which represents a specific shape which is projected.

6. A system as in claim 5, wherein said first part includes a number representing a prestored shape.

7. A system as in claim 5, wherein said first part includes information indicative of a basic shape, and information indicative of modifications to said basic shape.

8. A system as in claim 7, wherein said information indicative of said basic shape is a number representing a prestored shape.

9. A system as in claim 5, wherein said output signal also includes a second part which controls a specific color of said projected light.

10. A computer system comprising:

a computer based controller producing an output signal that includes parameters that digitally control a shape of a projected light; and an input to said computer based controller which allows an operator to control aspects of said shape.

11. A system as in claim 10, wherein said input allow the operator to select one of a plurality of different shapes.

12. A system as in claim 11, wherein said input further allows the operator to modify said one of said plurality of different shapes.

13. A system as in claim 12, wherein said user interface allows selecting one of a plurality of prestored shapes.

14. A system as in claim 13, wherein said user interface allows modifying said one of said prestored shapes.

15. A computer system comprising:

a computer based controller producing an output signal that includes parameters that digitally control a shape of a projected light; and a user interface, associated with said computer based controller, which user interface allows selecting said shapes.

16. A system, comprising:

a computer based controller which produces an output signal that allows digitally controlling at least a shape of a projected light, said output signal including information indicative of a basic shape of the projected light, and modifications to the basic shape.

17. A system as in claim 16, wherein said information indicative of the basic shape comprises a designation that represents the basic shape.

18. A system as in claim 16, wherein said information indicative of the basic shape is information that is indicative of a specified shape of unit size.

19. A system, comprising:

a computer based controller, including a user interface which allows designating a shape, and a controller which produces an output signal which is adapted to control a digital light to produce output light having the designated shape, wherein said output signal includes control information indicative of the designated shape, said control information including information indicative of a prestored shape.

20. A system as in claim 19, wherein said information indicative of the prestored shape comprises a designation which represents a specified unit shape.

21. A system as in claim 19, wherein said control information further includes information indicative of modification of said basic shape.

22. A system as in claim 19, wherein said output signal is a signal that is adapted to control a lighting device including a digital mirror device.

23. A system, comprising:

a user interface;

a computer based controller which is coupled to said user interface, and receives commands from said user interface, said commands including commands indicative of a shape of light to be projected, and said controller producing an output signal indicative of said commands.

24. A system as in claim 23, wherein said computer based controller further comprises a memory storing a plurality of different shapes for said light to be projected, each of said shapes including a designation associated therewith, and wherein said commands include information indicative of at least one of said designations.

25. A system as in claim 24, wherein said output signal also includes designations indicative of prestored shapes in an object being controlled by said output signal.

26. A system as in claim 24, further comprising at least one remote light, said light including a digital micromirror device therein enabling projection of light having a pre specified shape, and wherein the remote light includes shapes stored therein, and said commands include the designations indicative of at least one of said shapes.

27. A system as in claim 23, wherein said commands from said user interface also include a color of light to be projected, and said output signal includes light to be projected.

28. A system, comprising:

a computer based controller producing an output signal with a gobo control record therein, said gobo control record being a digital signal that specifies control of at least a shape and movement of an electronically-produced gobo.

29. A system as in claim 28, wherein the gobo control record includes a field representing a color of a part of the gobo part.

30. A system as in claim 28, wherein the gobo control record includes a field representing a type of the gobo being controlled.

31. A system, comprising;

a computer based controller producing an output signal with a gobo control record therein, said gobo control record being a digital signal that specifies control of at least a shape and movement of an electronically-produced gobo, wherein said gobo control records include a plurality of fields including at least shape of the gobo, movement of the gobo, and timing of the gobo.

32. A system, comprising:

a computer based controller producing an output signal with a gobo control record therein, said gobo control record being a digital signal that specifies control of at least a shape and movement of an electronically-produced gobo, wherein the gobo control record includes a field representing a thickness of a part of said gobo.

33. A system, comprising:

a computer based controller producing an output signal with a gobo control record therein, said gobo control record being a digital signal that specifies control of at least a shape and movement of an electronically-produced gobo, wherein the gobo control record includes a field representing a size of a part of the gobo.

34. A system, comprising:

a computer based controller producing an output signal with a gobo control record therein, said gobo control record being a digital signal that specifies control of at least a shape and movement of an electronically-produced gobo, wherein the gobo control record includes a field representing a position of a part of the gobo.

35. A system, comprising:

a computer based controller producing an output signal with a gobo control record therein, said gobo control record being a digital signal that specifies control of at least a shape and movement of an electronically-produced gobo, wherein the gobo record includes a field representing a type of the gobo being controlled, and wherein said type represents different ways that the gobo can be controlled.

36. A system as in claim 35, wherein said different ways include no moving parts or moving part.

37. A system as in claim 35, wherein said different ways include no moving parts or moving parts.

38. A system, comprising:

a computer based controller producing an output signal with a gobo control record therein, said gobo control record being a digital signal that specifies control of at least a shape and movement of an electronically-produced gobo, wherein the gobo record includes a field representing a type of the gobo being controlled, and wherein said type represents whether or not the gobo can be controlled.

39. A system, comprising:

a computer based controller producing an output signal with a gobo control record therein, said gobo control record including a gobo type which specifies the kinds of control to the gobo that can be represented by the record.

40. A system as in claim 39, wherein the gobo control record includes a field representing a thickness of a part of the gobo.

41. A system as in claim 39, wherein the gobo control record includes a field representing a size of a part of the gobo.

42. A system as in claim 39, wherein the gobo control record includes a field representing a position of a part of the gobo.

43. A system as in claim 39, wherein the gobo control record includes a field representing a color of a part of the gobo.

44. A system as in claim 39, wherein the gobo record includes a field representing a type of the gobo being controlled.

45. A system as in claim 44, wherein said type represents the different ways that the gobo can be controlled.

46. A system as in claim 44, wherein said type represents whether or not the gobo can be controlled.

47. A controller, comprising:

a controlling part which allows selecting one of a plurality of different gobo types to be controlled; and

a plurality of controlling encoders, associated with said controlling part, at least one of which encoders is assigned to control different values, depending on the gobo type being controlled.

48. A controller as in claim 47, further comprising a processor, which produces an output signal based on settings of said controlling part and said controlling encoders.

49. A controller as in claim 48, wherein said output signal represents a shape of light to be projected by a remote fixture.

50. A controller as in claim 48, wherein said output signal includes a first field for a control type, and for at least one of said controlled types, another field for information indicative of control variables.

51. A controller as in claim 47, wherein said controlling encoders control at least a time for gobo response.

52. A controller, comprising:

a controlling part which allows selecting one of a plurality of different gobos;

a console, with a plurality of controls, on said console, which are capable of being dynamically assigned, and

a processor, which determines said one of said plurality of different gobos, the values associated with said one of said plurality of different gobos, and assigns said plurality of controls based on the one of the plurality of different gobos which is selected.

53. A controller as in claim 52, wherein said plurality of controls include at least an encoder.

54. A controller as in claim 53, wherein said processor also produces an output signal that commands controlling the shape of projected light of a remote lighting fixture.

55. A controller, comprising:

a computer based controller producing an output signal that has a first part that commands a shape of a projected light and a second part that controls a time associated with said shape.

56. A controller as in claim 55, wherein said output signal commands changing between a first shape and a second shape, and said time represents a time to change from said first shape to said second shape.

57. A controller as in claim 55, wherein said output signal represents a control change, and said time represents a time that is allowed for the control change.

58. A controller as in claim 55, wherein said output signal represents the same gobo with different settings, and said time represents a time for move between the settings.

* * * * *

UNITED STATES PATENT AND TRADEMARK OFFICE

CERTIFICATE OF CORRECTION

PATENT NO. : 6,934,071 B2
DATED : August 23, 2005
INVENTOR(S) : Hunt

Page 1 of 1

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Title page,
Item [73], Assignee, delete "**Production Resource Group, L.L.P.**" insert -- **Production Resource Group, L.L.C.** --.

Signed and Sealed this

Twenty-seventh Day of December, 2005

Jon W. Dudas

JON W. DUDAS
Director of the United States Patent and Trademark Office

EXHIBIT S

US007020370B2

(12) **United States Patent**
Harris

(10) **Patent No.: US 7,020,370 B2**
(45) **Date of Patent: Mar. 28, 2006**

(54) **THREE COLOR DIGITAL GOBO SYSTEM**

(75) Inventor: **Jerry J. Harris**, Las Vegas, NV (US)

(73) Assignee: **Production Resource Group, L.L.C.**, New Windsor, NY (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **10/995,612**

(22) Filed: **Nov. 22, 2004**

(65) **Prior Publication Data**

US 2005/0100289 A1 May 12, 2005

Related U.S. Application Data

(63) Continuation of application No. 10/616,481, filed on Jul. 8, 2003, now Pat. No. 6,823,119, which is a continuation of application No. 09/771,953, filed on Jan. 29, 2001, now Pat. No. 6,588,944.

(51) **Int. Cl.**
G02B 6/44 (2006.01)
G02B 26/00 (2006.01)
G09F 13/00 (2006.01)

(52) **U.S. Cl.** **385/100**; 385/115; 385/116; 385/88; 385/147; 385/901; 359/291; 362/232; 362/551; 362/556

(58) **Field of Classification Search** 385/88, 385/89, 92, 49, 115, 116, 14, 147, 901, 37, 385/100; 359/291, 223, 224; 382/217, 220, 382/190; 348/241, 246, 239; 362/232, 551, 362/556, 293, 296
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,053,765	A	10/1991	Sonehara et al.	340/815.31
5,828,485	A	10/1998	Hewlett	359/291
5,940,204	A	8/1999	Hewlett	359/298
6,057,958	A	5/2000	Hunt	359/291
6,188,933	B1	2/2001	Hewlett et al.	700/19
6,208,087	B1	3/2001	Hughes et al.	315/291
6,256,136	B1	7/2001	Hunt	359/291
6,538,797	B1	3/2003	Hunt	359/291
6,588,944	B1	7/2003	Harris	385/88
6,617,792	B1 *	9/2003	Hughes et al.	315/32
6,823,119	B1 *	11/2004	Harris	385/100
6,891,656	B1 *	5/2005	Hunt	359/291

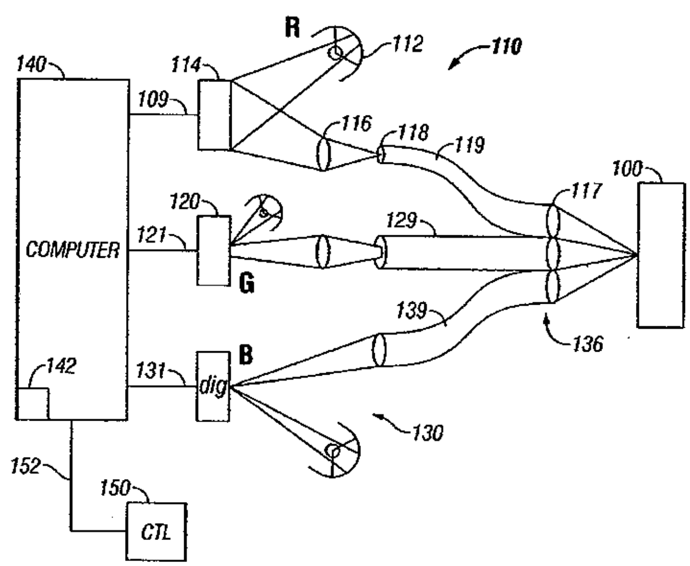
* cited by examiner

Primary Examiner—Brian M. Healy
(74) *Attorney, Agent, or Firm*—Fish & Richardson P.C.

(57) **ABSTRACT**

A system of digitally controlling light output by producing separate control signals for different colors of light. The light is contained in an optical waveguide, either prior to shaping or after shaping. Each of the control signals is coupled to a digitally controlled device which controls the shape of the light output. The digital controlling device can be digital mirror devices, for example.

104 Claims, 2 Drawing Sheets



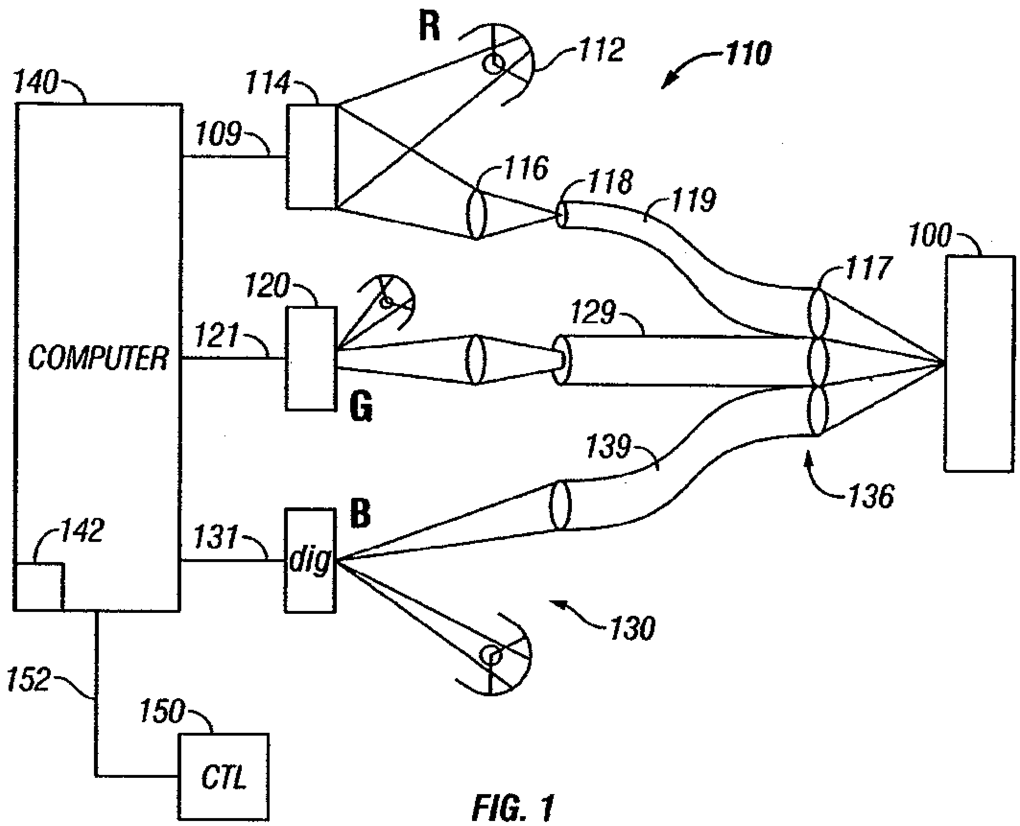

R
112
110
140
114
109
116
118
119
100
117
120
129
121
COMPUTER
G
139
136
142
B
131
dig
130
152
150
CTL

FIG. 1

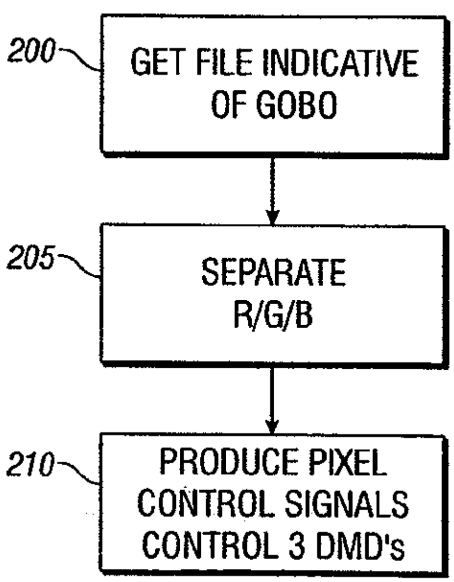

200
GET FILE INDICATIVE OF GOBO
205
SEPARATE R/G/B
210
PRODUCE PIXEL CONTROL SIGNALS CONTROL 3 DMD's

FIG. 2

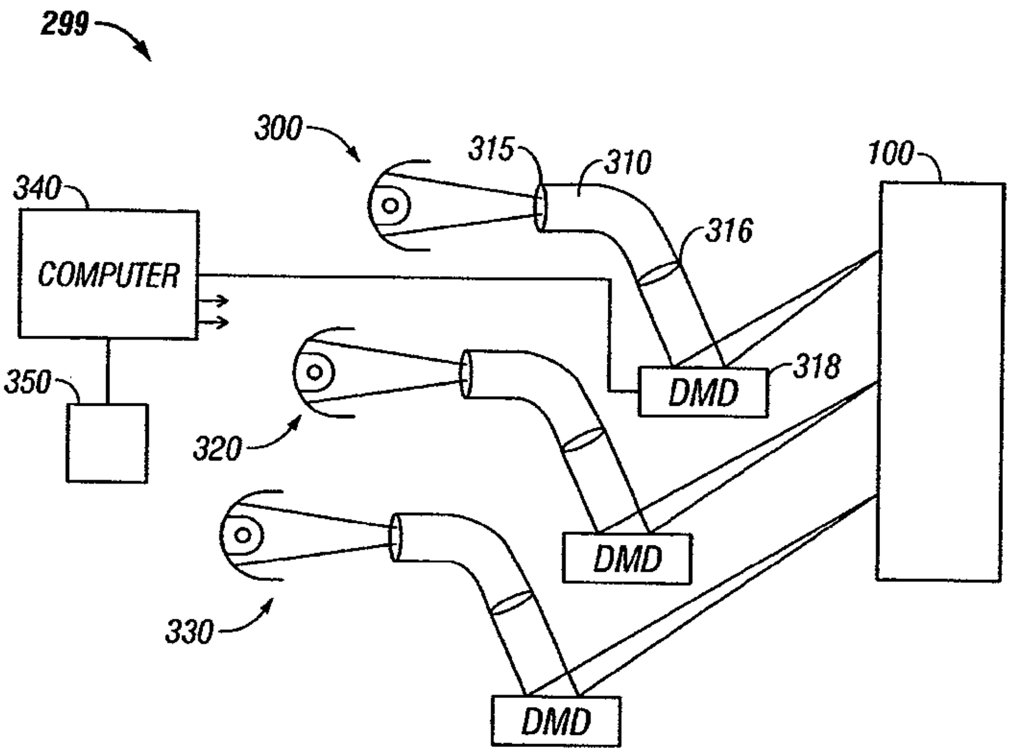

299
300
315
310
316
340
COMPUTER
100
318
DMD
350
320
DMD
330
DMD

FIG. 3

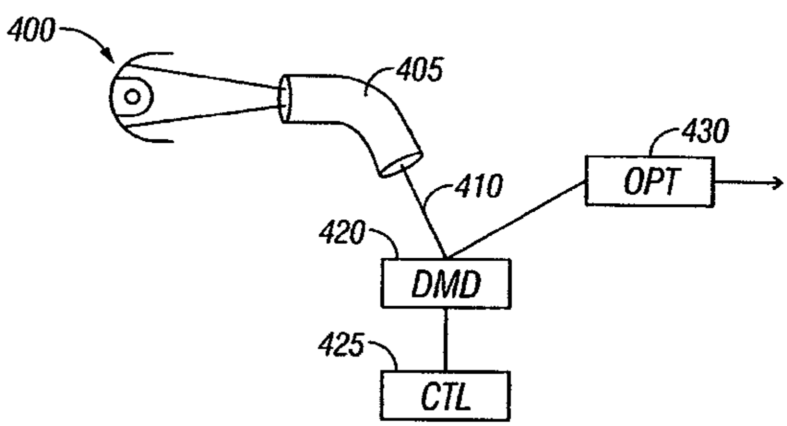

400
405
430
410
OPT
420
DMD
425
CTL

FIG. 4

THREE COLOR DIGITAL GOBO SYSTEM

CROSS-REFERENCE TO RELATED APPLICATIONS

This application is a continuation application of and claims priority to U.S. application Ser. No. 10/616,481, filed Jul. 8, 2003, now U.S. Pat. No. 6,823,119, which is a continuation of U.S. application Ser. No. 09/771,953, filed Jan. 29, 2001, now issued U.S. Pat. No. 6,588,944.

BACKGROUND

The U.S. Pat. No. 5,940,204 has suggested using a digital device to shape the contour and outlines of light that is projected through a high-intensity projector. Such a system may be used, for example, for stage lighting in theatrical and concert events. The Icon M™, available from Light and Sound Design, Ltd; Birmingham, England, uses this technique.

Different patents owned by Light and Sound Design, Ltd. suggest that the digital gobo should be formed from either a digital mirror, or from any other pixel level controllable digital device.

Cogent Light of Los Angeles, Calif. has technology that allows packaging a high intensity light beam into a form that allows it to be placed into a light waveguide, e.g., a fiber optic cable.

SUMMARY

The present application teaches a system of packaging light into a light waveguide such as a fiber optic cable, and adjusting the shape of the light using a digitally controllable, pixel level controllable light shaping element, such as a digital mirror device (DMD), available from Texas Instruments.

In one embodiment, the system controls and produces high-intensity light output using three separate digital gobo devices. The digital gobo devices can be separately controlled such that each digital gobo device receives information indicative of shaping a separate primary color. The primary colors are handled separately, and/or combined at the object of the high-intensity light output.

BRIEF DESCRIPTION OF THE DRAWINGS

These and other aspects will now be described in detail with reference to the accounts, wherein:

FIG. 1 is a block diagram of a three color version of the system; and

FIG. 2 shows a flowchart of operation of the controlling process for the digital gobo's in FIG. 1.

FIG. 3 shows a 3 DMD solution using three optical pipes;

FIG. 4 for shows a single DMD solution.

DETAILED DESCRIPTION

Details of a lighting instrument using a digital gobo are described in many patents owned by Light and Sound Design Ltd and the basic features are also present in Light and Sound Design's Icon M™ lighting fixture. The system described herein may use any of these basic features including details of computer-controlled cooling, and optics.

A block diagram of the basic system is shown in FIG. 1. An object of lighting **100** is shown. This object may be a stage, or may be any other object which is conventionally by

a high-intensity lighting device. The high-intensity lighting device may be, for example, a lighting device which produces more than 100 watts of lighting output, preferably more than 500 watts of lighting output. Devices of this type conventionally use a spotlight with a special high intensity bulb for producing the desired illumination effect.

In FIG. 1, three separate lighting units are formed. Each lighting unit is responsible for producing light of a separate primary color. The primary colors can be red, green and blue for additive colors, and cyan, magenta and yellow for subtractive coloration.

Each of the lighting units **110**, **120** and **130** are formed of similar structure. The lighting unit **110** includes a light source **112** which produces light of a specified primary color, here red. The lighting unit **110** may produce red coloration, or may include a white light with a red filter, or may even produce pure white light which is later filtered. The light from source **112** is applied to digital gobo device **114**. The digital gobo device **114** may be a digital mirror device available from Texas Instruments. Alternatively, the digital mirror device can be some other digitally controllable, pixel level controllable optical device such as, but not limited to, a grating light valve. The digital gobo device **114** is a controlling computer **140** which runs a specified program **142**. A controller **150** may be remote from the computer **140**, and connected to the computer by a line **152**. For example, the computer **140** may be within a separate lighting fixture along with the lighting elements **110**, **120** and **130**, and a remote central controller **150** may be a lighting control console.

The light output from the digital mirror device **114** is focused by an optics assembly **116**, and focused to the input end **118** of an optical waveguide **119**. The optical waveguide **119** may be, for example, a fiber-optic device including single or multiple fibers. The light input at end **119** is output at end **117**, and coupled towards the object **100**. Analogously, the other lighting unit **120** focuses its light onto a fiber-optic device **129**, and the lighting device **130** focuses its light onto a fiber-optic device **139**. Each of the lights may have different characteristics, i.e. they may have different coloration. The output of the three fiber-optic devices **119**, **129** and **139** are bundled together at area **136**, and are pointed towards the object of lighting **100**.

In this way, a number of advantages may be obtained. First, brighter light and different kinds of control may be obtained since the system disclosed herein uses three separate light sources. Moreover, better control over the digital gobo may be obtained since red; green and blue are separately controlled. Less flickering may be obtained, and more brightness, as compared with a system that uses only one DMD. Still a system that uses only one DMD is contemplated as described herein.

Different modifications on this system are possible. Other optical waveguides besides a fiber-optic pipe may be used in this system. Moreover, the optical filter which changes each of these separate light components to a separate light characteristic may be located after the digital mirror, e.g. as part of the optics assembly **116**, or on the input end of the fiber-optic device **118**.

The system is controlled according to the flowchart of FIG. 2. At **200**, a file indicative of a shaping of the light, e.g. a gobo to be used, is obtained. This file may be, for example, of the format described in U.S. Pat. No. 6,057,958. Of course, any file format can be used to define the gobo. The definition can be monochrome, gray scale, or full color (three different colors). At **205**, the file is changed to an image, and separated into its primary color components. In

the example given herein, the primary color components may include red, green and blue. Hence the file is separated into red, green and blue components. Such separation is conventional in video processing, and produces three separate signals. These three separate signals will eventually be used as the three separate controlling signals **109**, **121** and **131** respectively driving the red green and blue subassemblies. The control of the three separate digital mirror devices is carried out at **210**.

FIG. **3** shows an alternative embodiment which uses a similar concept. In the FIG. **3** embodiment, light is first launched from a light source **300** directly into a fiber-optic cable **310**. In this embodiment, the optics are shown as **315**, and are formed directly on the input end of the fiber-optic cable **310**. Light is launched into the fiber-optic cable, and hence may be focused and or colored by the optics **315**. Of course, this system may also use the separate optics shown as **116** in the FIG. **1** embodiment. Light is output on the output in **316** of the fiber-optic cable **310**, and coupled to a digital mirror device **318** which shapes the light and reflects it towards the object **100**.

The above has described a first channel shown as **299**. A separate second channel **320** produces a similar light alteration for the second aspect of light, while a third channel **330** produces a separate output for the third aspect of light; where the aspects can be colors. Each of the digital mirror devices may be controlled by the computer shown as **340** which may be controlled from a remote console **350**.

While the above has described control using three separate colors, it should be understood that two separate colors could also alternatively be used. Moreover, while the above describes the different aspects of light which are separately controlled being colors, it should be understood that any different aspect of shaping the beam of light could be separately controlled. For example, one alternative might use different intensity lights, each of which are separately controlled to produce some other kind of effect.

Another embodiment is shown in FIG. **4**. In this embodiment, a single DMD solution is shown. Light from the light **400** is immediately launched into an optical waveguide, e.g. fiber **405**. The fiber can be located in any configuration. It produces its light output **410** at the area of DMD **420**. As conventional, the DMD is controlled by a controller **425**. An optical assembly **430** receives the light from the DMD, and transmits it towards the object of illumination. The optical element **430** may include a color changing element therein, or multiple color changing elements, in order to produce full-color output. For example, the optical element **430** may include a spinning Red/Green/Blue filter which spins in synchronism with the changing of patterns on the DMD.

Although only a few embodiments have been disclosed in detail above, other modifications are possible. All such modifications are intended to be encompassed within the following claims, in which:

What is claimed is:

1. A method, comprising:

using a computer to process a digital file indicative of a color gobo shape to be projected;

producing outputs which are adapted to simultaneously control at least two digital light shape altering devices to states indicative of said color gobo shape for each of two different colors.

2. A method as in claim **1**, further comprising using said outputs to control at least two light shape altering devices.

3. A method as in claim **2**, further comprising using said light shape altering devices to alter a shape of light which is passed.

4. A method as in claim **3**, further comprising combining light which has been shaped by said at least two light shape altering devices into a composite light shape.

5. A method as in claim **3**, wherein said producing produces outputs for at least three light shape altering devices, and wherein said using comprises using said three light shape altering devices to shape a beam of light.

6. A method as in claim **4**, further comprising providing a first primary color of light to a first light shape altering device, and a second primary color of light to a second light shape altering device.

7. A method as in claim **2**, further comprising providing colored light to said light shape altering devices, wherein said providing comprises filtering white light to provide said colors of light.

8. A method as in claim **1**, wherein said producing simultaneously produces output signals respectively indicative of different color components of the color gobo shape.

9. A method as in claim **1**, wherein said digital light shape altering devices are digital mirror devices, and said outputs comprises two separate outputs which respectively drive two separate digital mirror devices.

10. A method as in claim **1**, further comprising projecting a light beam of an intensity of 500 watts or greater, based on said outputs.

11. A method as in claim **6**, wherein said providing comprises providing said light over fiber optic cables.

12. A method as in claim **1**, further comprising projecting a light beam that is shaped based on said digital file.

13. A method, comprising:

using a computer to process a digital file indicative of a composite light beam, that includes at least one shaped part therein, to be projected; and

producing outputs which are adapted to simultaneously control at least two digital light altering devices to different states, indicative of said composite light beam, and wherein at least one of said outputs represents a shape of said composite light beam.

14. A method as in claim **13**, further comprising using said outputs to control at least two light altering devices.

15. A method as in claim **14**, further comprising using at least one of said light altering devices to alter a shape of light which is passed.

16. A method as in claim **14**, further comprising providing light to said at least two light altering devices and, combining light output from said at least two light altering devices, into a composite light shape.

17. A method as in claim **14**, wherein said producing, produces outputs for at least three light altering devices, and wherein said using comprises using at least one of said three light altering devices to shape a beam of light.

18. A method as in claim **16**, wherein said providing comprises providing a first primary color of light to a first light altering device, and a second primary color of light to a second light altering device.

19. A method as in claim **14**, further comprising providing colored light to said light shape altering devices, wherein said providing comprises filtering white light to provide said colors of light.

20. A method as in claim **13**, wherein said producing simultaneously produces output signals respectively indicative of different color components of the color gobo shape.

21. A method as in claim **13**, wherein said light altering devices are digital mirror devices, and said outputs comprises two separate outputs which respectively drive two separate digital mirror devices.

22. A method as in claim 18, wherein said providing comprises providing said light over fiber optic cables.

23. A method, comprising:

using a computer to process at least one digital file indicative of a composite light beam formed of first and second image parts; and

producing outputs which are adapted to simultaneously control at least two digital, pixel-level controllable, light altering devices to different states, indicative of said composite light beam formed of both said first image part and said second image part.

24. A method as in claim 23, wherein at least one of said first and second image parts are an image of a gobo shape.

25. A method as in claim 23, further comprising using said outputs to control at least two light altering devices.

26. A method as in claim 23, further comprising providing light to said light shape altering devices, and combining light which has been modified by said at least two light altering devices into a composite light shape.

27. A method as in claim 23, wherein said producing, produces outputs for at least three light altering devices whose states collectively define the output light beam.

28. A method as in claim 23, further comprising providing a first color of light to a first light altering device, and a second color of light to a second light altering device.

29. A method as in claim 28, wherein said providing colored light comprises filtering white light to provide said colors of light.

30. A method as in claim 23, wherein said producing simultaneously produces output signals respectively indicative of different color components of the composite light beam.

31. A method as in claim 23, wherein said digital light altering devices are digital mirror devices, and said outputs comprises two separate outputs which respectively drive two separate digital mirror devices.

32. A method as in claim 28, wherein said providing comprises providing said light over fiber optic cables.

33. A method, comprising:

using a computer to process a digital file indicative of a composite light beam formed of first and second image parts; and

producing outputs which are adapted to simultaneously control at least two pixel level controllable digital light altering devices to different states, indicative of said composite light beam, and wherein said outputs represents states of image portions of different intensities.

34. A method as in claim 33, further comprising using said outputs to control at least two light altering devices.

35. A method as in claim 34, further comprising using said light altering devices to alter a shape of light which is passed.

36. A method as in claim 35, further comprising combining light which has been shaped by said at least two light altering devices into a composite light beam.

37. A method as in claim 35, wherein said producing, produces outputs for at least three light altering devices, and wherein said using comprises using said three light altering devices to form said composite beam of light.

38. A method as in claim 36, further comprising providing a first intensity color of light to a first light altering device, and a second intensity of light to a second light altering device.

39. A method as in claim 33, wherein said producing simultaneously produces output signals respectively indicative of different intensity components of the composite beam of light.

40. A method as in claim 33, wherein said digital light altering devices are digital mirror devices, and said outputs comprises three separate outputs which respectively drive two separate digital mirror devices.

41. A method as in claim 38, wherein said providing comprises providing said light over fiber optic cables.

42. A method comprising:

obtaining a signal to be used to shape a composite beam of light;

separating said signal into at least first and second different signals, at least one of which controls shaping a portion of said composite beam of light; and

providing said first and second signals simultaneously for use in shaping said light.

43. A method as in claim 42, further comprising using said first and second signals to simultaneously drive electronic devices which simultaneously produce first and second light parts.

44. A method as in claim 42, further comprising illuminating said electronic devices with light, and combining outputs of said electronic devices.

45. A method as in claim 42, wherein said aspect of light is light colors, and said first and second different signals simultaneously control different light colors.

46. A method as in claim 44, wherein said illuminating comprises using a fiber optic element to illuminate said electronic devices.

47. An apparatus, comprising:

means for forming first, second and third color components, each of which represents a primary color component of a desired shape and color for light; and

means for producing controlling signals to drive three digitally controllable light shape altering devices with the first, second and third color components, simultaneously.

48. An apparatus as in claim 47, further comprising digital mirror devices driven by color components.

49. An apparatus as in claim 47, wherein said primary color components comprise red, green and blue color components.

50. An apparatus comprising:

a computer which processes a digital file indicative of a color gobo shape to be projected; and

produces at least one output in a format to simultaneously and separately control at least two digital light shape altering devices to respective states indicative of said color gobo shape for each of two different colors.

51. An apparatus as in claim 50, further comprising first and second light shape altering devices, receiving said at least one output and each of said devices altering a shape of light which is passed thereby.

52. An apparatus as in claim 51, further comprising a light combining system that combines light which has been shaped by said first and second light shape altering devices into a composite light shape.

53. An apparatus as in claim 51, further comprising a third light shape altering device, and wherein said at least one output controls said third light shape altering device.

54. An apparatus as in claim 52, further comprising a first light source providing a first primary color of light to said first light shape altering device, and a second light source providing a second primary color of light to said second light shape altering device.

55. An apparatus as in claim 50, wherein said digital light shape altering devices are digital mirror devices, and said at least one output comprises two separate outputs which respectively drive two separate digital mirror devices.

56. An apparatus as in claim 51, wherein said providing comprises providing said light over fiber optic cables.

57. An apparatus as in claim 50, further comprising a light projecting system, projecting a light beam that is shaped based on said digital file.

58. An apparatus, comprising:

a computer, producing a digital file indicative of a desired composite light beam that includes at least one shaped part therein, and producing outputs which are adapted to simultaneously control at least two digital light altering devices to different states indicative of said composite light beam, and wherein at least one of said outputs represents a shape of said composite light beam.

59. An apparatus as in claim 58, further comprising, at least two light altering devices, controlled by said outputs.

60. An apparatus as in claim 59, wherein at least one of said light altering devices is operable to alter a shape of light which is passed thereby.

61. An apparatus as in claim 59, further comprising at least one light source, illuminating said at least two light shape altering devices and a light combining part, combining light output from said at least two light altering devices into a composite light shape.

62. An apparatus as in claim 59, further comprising a third light altering device, and wherein said using comprises using at least one of said first, second and/or third light altering devices to shape a beam of light which forms a portion of a composite beam.

63. An apparatus as in claim 61, wherein said at least one light source includes a first light source providing a first primary color of light to said first light shape altering device, and a second light source providing a second primary color of light to said second light altering device.

64. An apparatus as in claim 58, wherein said light altering devices are digital mirror devices, and said outputs comprises two separate outputs which respectively drive two separate digital mirror devices.

65. An apparatus as in claim 61, further comprising fiber optic cables providing light from said at least one light source to said light altering devices.

66. An apparatus, comprising:

a computer that processes at least one digital file indicative of a composite light beam formed of first and second image parts, and producing outputs which are formatted to simultaneously control at least two digital, pixel-level controllable, light altering devices to different states, indicative of said composite light beam formed of both said first image part and said second image part.

67. An apparatus as in claim 66, wherein at least one of said first and second image parts is an image of a gobo shape.

68. An apparatus as in claim 66, further comprising at least two light altering devices, controlled by said outputs.

69. An apparatus as in claim 66, further comprising a light source, providing light to said light shape altering devices, and a light combining element, combining light which has been modified by said at least two light altering devices into a composite light shape.

70. An apparatus as in claim 66, further comprising a third light altering device controlled by said outputs.

71. An apparatus as in claim 69, wherein said at least one light source further comprises a first light source providing a first color of light to said first light altering device, and a second light source providing a second color of light to a second light altering device.

72. An apparatus as in claim 66, wherein said digital light altering devices are digital mirror devices, and said outputs comprises two separate outputs which respectively drive two separate digital mirror devices.

73. An apparatus as in claim 69, further comprising fiber optic cables, providing said light from said at least one light source, to said light altering devices.

74. An apparatus, comprising:

a computer that produces a digital file indicative of a composite light beam formed of first and second image parts, and which produces outputs which are adapted to simultaneously control at least two pixel level controllable digital light altering devices to different states, indicative of said composite light beam, and wherein said outputs respectively represent states of light beams of different intensities that collectively form said composite light beam.

75. An apparatus as in claim 74, further comprising at least two light altering devices controlled by said outputs.

76. An apparatus as in claim 75, wherein at least one of said first and second image parts defines a shape of light which is passed.

77. An apparatus as in claim 76, further comprising a light combining element that combines light which has been shaped by said at least two light shape altering devices into a composite light beam.

78. An apparatus as in claim 74, further comprising a third light altering device.

79. An apparatus as in claim 75, further comprising a first light source providing a first intensity color of light to said first light altering device, and a second light source providing a second intensity of light to said second light altering device.

80. An apparatus as in claim 74, wherein said output signals respectively indicative of portions representing different intensity components of the composite light beam.

81. An apparatus as in claim 75, wherein said digital light altering devices are digital mirror devices, and said outputs comprises three separate outputs which respectively drive two separate digital mirror devices.

82. An apparatus as in claim 79, wherein said providing comprises providing said light over fiber optic cables.

83. A apparatus, comprising:

means for processing a digital file indicative of a color gobo shape to be projected, said processing means including means for producing outputs which are adapted to simultaneously control at least two digital light shape altering devices to states indicative of said color gobo shape for each of two different colors.

84. An apparatus as in claim 83, further comprising at least two light shape altering means for changing applied light based on said outputs.

85. An apparatus as in claim 84, further comprising means for projecting light to each of said at least two light shape altering means.

86. An apparatus as in claim 84, further comprising means for combining light which has been shaped by said at least two light shape altering means into a composite light shape.

87. An apparatus as in claim 85, wherein said projecting means comprises means for providing a first primary color of light to a first light shape altering means, and means for providing a second primary color of light to a second light shape altering means.

88. An apparatus as in claim 84, wherein said digital light shape altering means include digital mirror devices, and said outputs comprises two separate outputs which respectively drive two separate digital mirror devices.

89. An apparatus, comprising:

means for processing a digital file indicative of a composite light beam, that includes at least one shaped part therein, said processing means comprising means for producing outputs which are adapted to simultaneously control at least two digital light altering devices to different states, indicative of said composite light beam, and wherein at least one of said outputs represents a shape of said composite light beam.

90. A apparatus as in claim 89, at least two light altering means for changing an applied light beam, based on said outputs.

91. A apparatus as in claim 90, further comprising means for providing light to said at least two light altering devices and means for combining light output from said at least two light altering devices, into a composite light shape.

92. An apparatus as in claim 91, wherein said providing means comprises first means for providing a first color of light to a first light altering device, and a second primary color of light to a second light altering device.

93. An apparatus as in claim 89, wherein said light altering means are digital mirror devices, and said outputs comprise two separate outputs which respectively drive two separate digital mirror devices.

94. An apparatus, comprising:

means for processing at least one digital file indicative of a composite light beam formed of first and second image parts, said processing means including means for producing outputs which are adapted to simultaneously control at least two digital, pixel-level controllable, light altering devices to different states, indicative of said composite light beam formed of both said first image part and said second image part.

95. An apparatus as in claim 94, wherein at least one of said first and second image parts are an image of a gobo shape.

96. An apparatus as in claim 94, further comprising at least two light altering means for changing a light beam shape based on said outputs.

97. An apparatus as in claim 94, further comprising means for providing light to said light shape altering means, and means for combining light which has been modified by said at least two light altering means into a composite light shape.

98. An apparatus as in claim 97, wherein said providing means comprises first means for providing a first color of light to a first light altering device, and second means for providing a second color of light to a second light altering device.

99. A apparatus as in claim 94, wherein said digital light altering means include digital mirror devices, and said outputs comprises two separate outputs which respectively drive two separate digital mirror devices.

100. An apparatus, comprising:

means for processing a digital file indicative of a composite light beam formed of first and second image parts, said processing means comprising means for producing outputs which are adapted to simultaneously control at least two pixel level controllable digital light altering devices to different states, indicative of said composite light beam, and wherein said outputs represents states of image portions of different intensities.

101. An apparatus as in claim 100, further comprising at least two light altering means for changing an aspect of applied light based on said outputs.

102. An apparatus as in claim 101, further comprising means for projecting light to said at least two light altering means.

103. An apparatus as in claim 102, further comprising means for combining light which has been shaped by said at least two light altering means into a composite light beam.

104. An apparatus as in claim 101, wherein said projecting means further comprises first means for providing a first intensity color of light to a first light altering means and second means for providing a second intensity of light to a second light altering means.

* * * * *

EXHIBIT T

US007181112B2

(12) United States Patent
Harris

(10) Patent No.: US 7,181,112 B2
(45) Date of Patent: Feb. 20, 2007

(54) **THREE COLOR DIGITAL GOBO SYSTEM**

(75) Inventor: **Jeremiah J. Harris**, Las Vegas, NV (US)

(73) Assignee: **Production Resource Group, L.L.C.**, New Windsor, NY (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **11/386,194**

(22) Filed: **Mar. 21, 2006**

(65) **Prior Publication Data**

US 2006/0177185 A1 Aug. 10, 2006

Related U.S. Application Data

(63) Continuation of application No. 10/995,612, filed on Nov. 22, 2004, now Pat. No. 7,020,370, which is a continuation of application No. 10/616,481, filed on Jul. 8, 2003, now Pat. No. 6,823,119, which is a continuation of application No. 09/771,953, filed on Jan. 29, 2001, now Pat. No. 6,588,944.

(51) **Int. Cl.**
G02B 6/44 (2006.01)
G02B 26/00 (2006.01)
G09F 13/00 (2006.01)

(52) **U.S. Cl.** **385/100**; 385/115; 385/116; 385/88; 385/147; 385/901; 359/291; 362/232; 362/551; 362/556

(58) **Field of Classification Search** 385/88, 385/89, 92, 49, 115, 116, 14, 147, 901, 37, 385/100; 359/291, 223, 224; 382/217, 220, 382/190; 348/241, 239, 246; 362/232, 551, 362/556, 293, 296
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,053,765	A	10/1991	Sonehara et al.	340/815.31
5,774,174	A *	6/1998	Hardie	348/38
5,828,485	A	10/1998	Hewlett	339/291
5,940,204	A	8/1999	Hewlett	359/298
6,057,958	A	5/2000	Hunt	359/291
6,188,933	B1	2/2001	Hewlett et al.	700/19
6,208,087	B1	3/2001	Hughes et al.	315/291
6,256,136	B1	7/2001	Hunt	359/291
6,538,797	B1	3/2003	Hunt	359/291
6,588,944	B2	7/2003	Harris	385/88
6,617,792	B2	9/2003	Hughes et al.	315/32
6,823,119	B2	11/2004	Harris	385/100
6,891,656	B2	5/2005	Hunt	359/291
2002/0181070	A1 *	12/2002	Hewlett	359/291
2005/0100289	A1 *	5/2005	Harris	385/88
2006/0177185	A1 *	8/2006	Harris	385/100

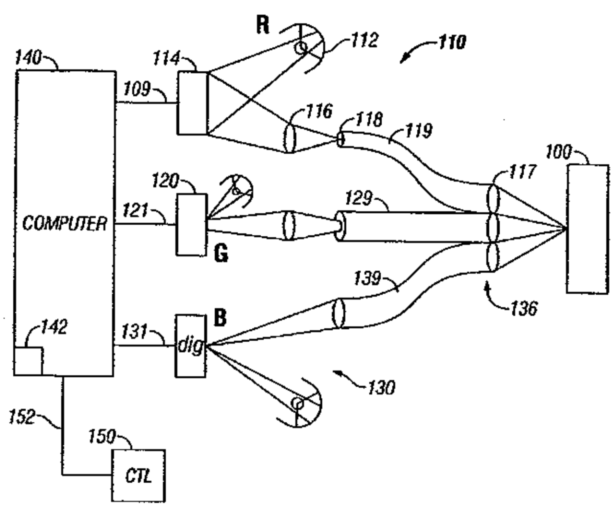
* cited by examiner

Primary Examiner—Brian M. Healy
(74) *Attorney, Agent, or Firm*—Fish & Richardson P.C.

(57) **ABSTRACT**

A system of digitally controlling light output by producing separate control signals for different colors of light. The light is contained in an optical waveguide, either prior to shaping or after shaping. Each of the control signals is coupled to a digitally controlled device which controls the shape of the light output. The digital controlling device can be digital mirror devices, for example.

45 Claims, 2 Drawing Sheets



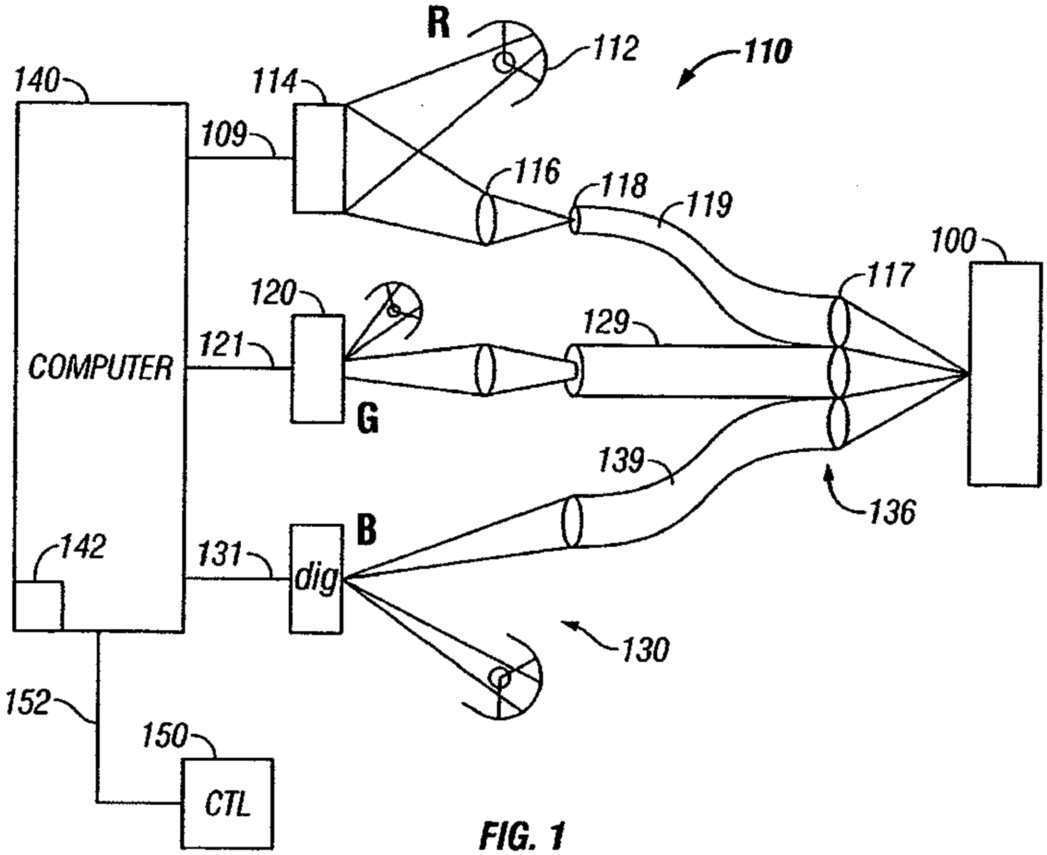

R
112
110
140
114
109
116
118
119
100
117
120
129
121
COMPUTER
G
139
136
B
142
131
dig
130
152
150
CTL

FIG. 1

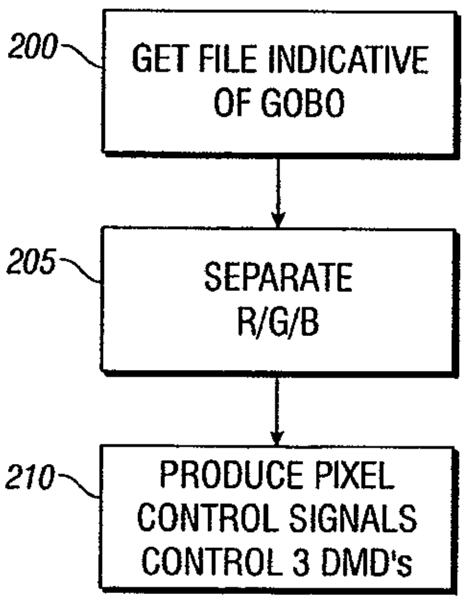
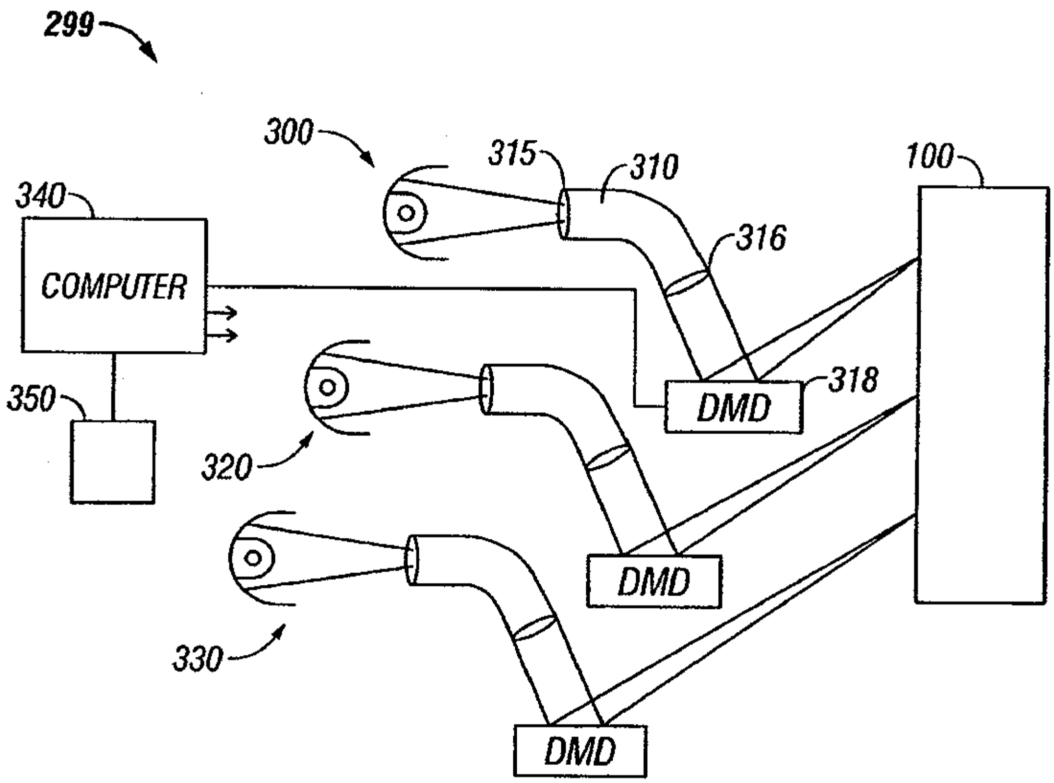
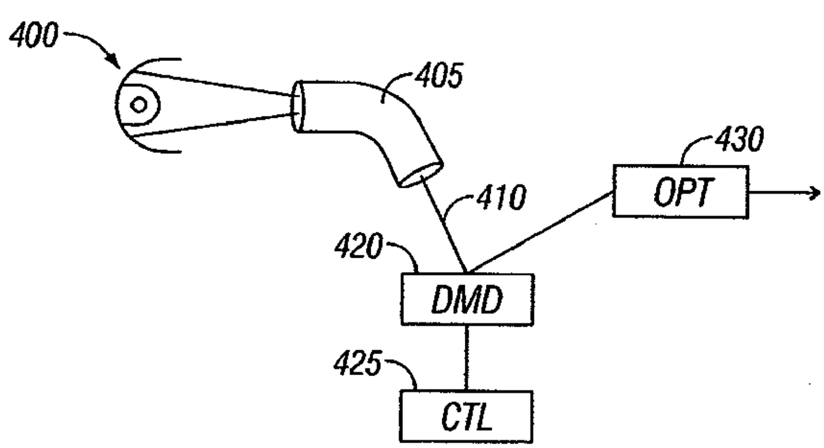

200
GET FILE INDICATIVE
OF GOBO
205
SEPARATE
R/G/B
210
PRODUCE PIXEL
CONTROL SIGNALS
CONTROL 3 DMD's

FIG. 2



299
300
315
310
316
100
340
COMPUTER
350
320
DMD
318
DMD
330
DMD

FIG. 3



400
405
430
410
OPT
420
DMD
425
CTL

FIG. 4

THREE COLOR DIGITAL GOBO SYSTEM

CROSS-REFERENCE TO RELATED APPLICATIONS

This application is a continuation application of and claims priority to U.S. application Ser. No. 10/995,612, filed Nov. 22, 2004, now U.S. Pat. No. 7,020,370, which is a continuation of U.S. application Ser. No. 10/616,481, filed Jul. 8, 2003, now U.S. Pat. No. 6,823,119, which is a continuation of U.S. application Ser. No. 09/771,953, filed Jan. 29, 2001, now U.S. Pat. No. 6,588,944.

BACKGROUND

The U.S. Pat. No. 5,940,204 has suggested using a digital device to shape the contour and outlines of light that is projected through a high-intensity projector. Such a system may be used, for example, for stage lighting in theatrical and concert events. The Icon M™, available from Light and Sound Design, Ltd; Birmingham, England, uses this technique.

Different patents owned by Light and Sound Design, Ltd. suggest that the digital gobo should be formed from either a digital mirror, or from any other pixel level controllable digital device.

Cogent Light of Los Angeles, Calif. has technology that allows packaging a high intensity light beam into a form that allows it to be placed into a light waveguide, e.g., a fiber optic cable.

SUMMARY

The present application teaches a system of packaging light into a light waveguide such as a fiber optic cable, and adjusting the shape of the light using a digitally controllable, pixel level controllable light shaping element, such as a digital mirror device (DMD), available from Texas Instruments.

In one embodiment, the system controls and produces high-intensity light output using three separate digital gobo devices. The digital gobo devices can be separately controlled such that each digital gobo device receives information indicative of shaping a separate primary color. The primary colors are handled separately, and/or combined at the object of the high-intensity light output.

BRIEF DESCRIPTION OF THE DRAWINGS

These and other aspects will now be described in detail with reference to the accounts, wherein:

FIG. 1 is a block diagram of a three color version of the system.

FIG. 2 shows a flowchart of operation of the controlling process for the digital gobo's in FIG. 1.

FIG. 3 shows a 3 DMD solution using three optical pipes.

FIG. 4 for shows a single DMD solution.

DETAILED DESCRIPTION

Details of a lighting instrument using a digital gobo are described in many patents owned by Light and Sound Design Ltd and the basic features are also present in Light and Sound Design's Icon M™ lighting fixture. The system described herein may use any of these basic features including details of computer-controlled cooling, and optics.

A block diagram of the basic system is shown in FIG. 1. An object of lighting 100 is shown. This object may be a stage, or may be any other object which is conventionally by a high-intensity lighting device. The high-intensity lighting device may be, for example, a lighting device which produces more than 100 watts of lighting output, preferably more than 500 watts of lighting output. Devices of this type conventionally use a spotlight with a special high intensity bulb for producing the desired illumination effect.

In FIG. 1, three separate lighting units are formed. Each lighting unit is responsible for producing light of a separate primary color. The primary colors can be red, green and blue for additive colors, and cyan, magenta and yellow for subtractive coloration.

Each of the lighting units 110, 120 and 130 are formed of similar structure. The lighting unit 110 includes a light source 112 which produces light of a specified primary color, here red. The lighting unit 110 may produce red coloration, or may include a white light with a red filter, or may even produce pure white light which is later filtered. The light from source 112 is applied to digital gobo device 114. The digital gobo device 114 may be a digital mirror device available from Texas Instruments. Alternatively, the digital mirror device can be some other digitally controllable, pixel level controllable optical device such as, but not limited to, a grating light valve. The digital gobo device 114 is a controlling computer 140 which runs a specified program 142. A controller 150 may be remote from the computer 140, and connected to the computer by a line 152. For example, the computer 140 may be within a separate lighting fixture along with the lighting elements 110, 120 and 130, and a remote central controller 150 may be a lighting control console.

The light output from the digital mirror device 114 is focused by an optics assembly 116, and focused to the input end 118 of an optical waveguide 119. The optical waveguide 119 may be, for example, a fiber-optic device including single or multiple fibers. The light input at end 119 is output at end 117, and coupled towards the object 100. Analogously, the other lighting unit 120 focuses its light onto a fiber-optic device 129, and the lighting device 130 focuses its light onto a fiber-optic device 139. Each of the lights may have different characteristics, i.e. they may have different coloration. The output of the three fiber-optic devices 119, 129 and 139 are bundled together at area 136, and are pointed towards the object of lighting 100.

In this way, a number of advantages may be obtained. First, brighter light and different kinds of control may be obtained since the system disclosed herein uses three separate light sources. Moreover, better control over the digital gobo may be obtained since red; green and blue are separately controlled. Less flickering may be obtained, and more brightness, as compared with a system that uses only one DMD. Still a system that uses only one DMD is contemplated as described herein.

Different modifications on this system are possible. Other optical waveguides besides a fiber-optic pipe may be used in this system. Moreover, the optical filter which changes each of these separate light components to a separate light characteristic may be located after the digital mirror, e.g. as part of the optics assembly 116, or on the input end of the fiber-optic device 118.

The system is controlled according to the flowchart of FIG. 2. At 200, a file indicative of a shaping of the light, e.g. a gobo to be used, is obtained. This file may be, for example, of the format described in U.S. Pat. No. 6,057,958. Of course, any file format can be used to define the gobo. The

definition can be monochrome, gray scale, or full color (three different colors). At 205, the file is changed to an image, and separated into its primary color components. In the example given herein, the primary color components may include red, green and blue. Hence the file is separated into red, green and blue components. Such separation is conventional in video processing, and produces three separate signals. These three separate signals will eventually be used as the three separate controlling signals 109, 121 and 131 respectively driving the red green and blue subassemblies. The control of the three separate digital mirror devices is carried out at 210.

FIG. 3 shows an alternative embodiment which uses a similar concept. In the FIG. 3 embodiment, light is first launched from a light source 300 directly into a fiber-optic cable 310. In this embodiment, the optics are shown as 315, and are formed directly on the input end of the fiber-optic cable 310. Light is launched into the fiber-optic cable, and hence may be focused and or colored by the optics 315. Of course, this system may also use the separate optics shown as 116 in the FIG. 1 embodiment. Light is output on the output in 316 of the fiber-optic cable 310, and coupled to a digital mirror device 318 which shapes the light and reflects it towards the object 100.

The above has described a first channel shown as 299. A separate second channel 320 produces a similar light alteration for the second aspect of light, while a third channel 330 produces a separate output for the third aspect of light; where the aspects can be colors. Each of the digital mirror devices may be controlled by the computer shown as 340 which may be controlled from a remote console 350.

While the above has described control using three separate colors, it should be understood that two separate colors could also alternatively be used. Moreover, while the above describes the different aspects of light which are separately controlled being colors, it should be understood that any different aspect of shaping the beam of light could be separately controlled. For example, one alternative might use different intensity lights, each of which are separately controlled to produce some other kind of effect.

Another embodiment is shown in FIG. 4. In this embodiment, a single DMD solution is shown. Light from the light 400 is immediately launched into an optical waveguide, e.g. fiber 405. The fiber can be located in any configuration. It produces its light output 410 at the area of DMD 420. As conventional, the DMD is controlled by a controller 425. An optical assembly 430 receives the light from the DMD, and transmits it towards the object of illumination. The optical element 430 may include a color changing element therein, or multiple color changing elements, in order to produce full-color output. For example, the optical element 430 may include a spinning Red/Green/Blue filter which spins in synchronism with the changing of patterns on the DMD.

Although only a few embodiments have been disclosed in detail above, other modifications are possible. All such modifications are intended to be encompassed within the following claims, in which:

What is claimed is:

1. A method, comprising: obtaining an electronic file representative of a color gobo shape;

dividing said electronic file into a first part representative of a first part of the color gobo shape, and into a second part representative of a second part of the color gobo shape, where the first part and second parts collectively represent a complete color gobo shape; and

using said first part to control a first part of shaping a light according to the color gobo shape, and using said second part to control a second part of shaping the light according to the color gobo shape.

2. A method as in claim 1, wherein said first part is a first color component of the light shaping by the color gobo shape, and said second part is a second color component of the light shaping by the color gobo shape.

3. A method as in claim 1, further comprising dividing said electronic file into an additional third part representative of a third part of the color gobo shape.

4. A method as in claim 3, wherein said first part is a first color component, said second part is a second color component, and said third part is a third color component of the light shaping by the color gobo shape.

5. A method as in claim 4, wherein said colors components are red, green and blue respectively.

6. A method as in claim 4, wherein said color components are cyan magenta and yellow respectively.

7. A method as in claim 1, wherein said using comprises digitally controlling shaping of the light.

8. A method as in claim 1, wherein said using comprises digitally controlling shaping the light in separate digitally controllable pixel level light shape altering devices.

9. A method as in claim 1, wherein said obtaining an electronic file comprises using a computer to process a digital file which is stored in said computer.

10. A method as in claim 8, further comprising using said digitally controllable pixel light shape altering devices to shape at least one beam of light.

11. A method, comprising:

forming first and second electronic files, respectively indicative of first and second component parts, which first and second component parts collectively represent a color gobo that defines shaping of an outer perimeter of a beam of light, and that defines a color within the shaped outer perimeter; and

using both said first and second component parts to form the color gobo in the specified shape with the shaped outer perimeter, and with the defined color within the shaped outer perimeter, further comprising forming a third electronic file indicative of a third component part, and where said first, second and third component parts collectively represent the color gobo, and wherein said using comprises using all of the first, second and third component parts to form the color gobo.

12. A method as in claim 11, wherein said using comprises applying the first component part to a first digitally controllable light shape altering device, applying the second component part to a second digitally controllable light shape altering device.

13. A method as in claim 12, further comprising applying a first light beam to said first digital light shape altering device, applying a second light beam to said second digital light shape altering device, and combining outputs of said first and second light shape altering devices to form the color gobo.

14. A method as in claim 11, wherein said first and second component parts represent different color components of the color gobo.

15. A method, comprising:

forming first and second electronic files, respectively indicative of first and second component parts, which first and second component parts collectively represent a color gobo that defines shaping of an outer perimeter of a beam of light, and that defines a color within the shaped outer perimeter; and

using both said first and second component parts to form the color gobo in the specified shape with the shaped

outer perimeter, and with the defined color within the shaped outer perimeter, wherein said forming comprises obtaining an electronic file indicative of a color gobo, and separating said electronic file into said first and second component parts.

16. A method, comprising:

separately shaping outer perimeters of each of first and second separate light beams; and

combining said first and second light beams with shaped outer perimeters into a composite light beam, wherein said combining comprises using a fiber-optic device to combine said light beam.

17. A method, comprising:

separately shaping outer perimeters of each of first and second separate light beams; and

combining said first and second light beams with shaped outer perimeters into a composite light beam, wherein said shaping comprises shaping said outer perimeters according to first and second component files, which first and second component files collectively represent a color gobo that shapes the outer perimeter of a beam of light and colors a section within said shaped outer perimeter.

18. A method as in claim **17**, wherein said shaping comprises obtaining a file indicative of a color gobo, and separating said file into said first and second component files.

19. A method as in claim **18**, wherein said first and second component files respectively represent primary color portions of the color gobo.

20. A method as in claim **16**, further comprising using a first light source to create said first light beam and using a second light source to create said second light beam.

21. A method, comprising:

obtaining a color gobo electronic file representing a color gobo;

using said color gobo electronic file to form first, second and third electronic files, respectively indicative of first, second and third component parts, which first, second and third component parts collectively represent a color gobo that defines a shape of an outer perimeter of a beam of light, and that defines at least one color within the shaped outer perimeter;

obtaining first, second and third light beams;

separately shaping outer perimeters of each of said first, second and third light beams; and

combining said first, second and third light beams with shaped outer perimeters into a composite light beam.

22. A method as in claim **21**, wherein said first, second and third light beams are each in different primary colors.

23. An apparatus, comprising:

a computer, obtaining an electronic file representative of a color gobo shape, and dividing said electronic file into a first part representative of a first part of the color gobo shape, and into a second part representative of a second part of the color gobo shape, where the first part and second parts collectively represent a complete color gobo shape; and

an optical part, using said first part to control a first part of shaping a light according to the color gobo shape, and using said second part to control a second part of shaping the light according to the color gobo shape.

24. An apparatus as in claim **23**, wherein said first part is a first color component of the light shaping by the color gobo shape, and said second part is a second color component of the light shaping by the color gobo shape.

25. An apparatus as in claim **23**, wherein said computer further divides said electronic file into an additional third part representative of a third part of the color gobo shape.

26. An apparatus as in claim **25**, wherein said first part is a first color component, said second part is a second color component, and said third part is a third color component of the light shaping by the color gobo shape.

27. An apparatus as in claim **26**, wherein said color components are red, green and blue respectively.

28. An apparatus as in claim **26**, wherein said color components are cyan magenta and yellow respectively.

29. An apparatus as in claim **23**, wherein said optical part includes a first digitally controllable light shape altering device, controlled by said first part, and a second digitally controllable light shape altering device, controlled by said second part.

30. An apparatus as in claim **25**, wherein said optical part includes a first digitally controllable light shape altering device, controlled by said first part, and a second digitally controllable light shape altering device, controlled by said second part and a third digitally controllable light shape altering device, controlled by said third part.

31. An apparatus as in claim **30**, further comprising three light sources, producing light in each of three primary colors.

32. An apparatus, comprising:

sa computer, operating to form first and second electronic files, respectively indicative of first and second component parts, which first and second component parts collectively represent a color gobo that defines shaping of an outer perimeter of a beam of light, and that defines a color within the shaped outer perimeter; and

an optical system, receiving said first and second electronic files, and forming the color gobo in the specified shape with the shaped outer perimeter, and with the defined color within the shaped outer perimeter based on said first and second optical files.

33. An apparatus as in claim **32**, wherein said computer further forms a third electronic file indicative of a third component part, and where said first, second and third component parts collectively represent the color gobo, and wherein said is responsive to each of the first, second and third electronic files to form the color gobo.

34. An apparatus as in claim **32**, wherein said optical system comprises a first digital light shape altering device receiving said first electronic file, and a second digital light shape altering device receiving the second electronic file.

35. An apparatus as in claim **34**, wherein said digital light shape altering devices are digital mirror devices.

36. An apparatus as in claim **34**, further comprising a first light source, applying a first light beam to said first digital light shape altering device, and a second light source applying a second light beam to said second digital light shape altering device, and an optical combiner, combining outputs of said first and second light shape altering devices to form the color gobo.

37. An apparatus as in claim **36**, wherein said first light source produces a first color light, and said second light source produces a second color light.

38. An apparatus as in claim **32**, wherein said computer processes an electronic file indicative of a color gobo to form said first and second component parts representative of primary color portions within said electronic file.

39. An apparatus, comprising: first and second light sources, producing first and second separate light beams;

first and second electronically controllable gobo devices, each separately shaping outer perimeters of each of first and second separate light beams; and

a combiner, combining said first and second light beams with shaped outer perimeters into a composite light beam.

40. An apparatus as in claim **39**, further comprising a third light source, forming a third light beam, and a third electronically controllable gobo device, separately shaping said outer perimeter of said third light beam, and wherein said combiner combines said first, second and third light beams.

41. An apparatus as in claim **39**, wherein said first and second light beams are of different colors.

42. An apparatus as in claim **39**, wherein said combiner comprises a fiber-optic device.

43. An apparatus as in claim **39**, wherein said first and second light sources produce light of different primary colors.

44. An apparatus, comprising:

a computer that stores a color gobo electronic file representing a color gobo and operates using said color gobo electronic file to form first, second and third electronic files, respectively indicative of first, second and third component parts, which first, second and third component parts collectively represent a color gobo that defines a shape of an outer perimeter of a beam of light, and that defines at least one color within the shaped outer perimeter;

first, second and third light sources, respectively producing first, second and third light beams;

first, second and third electronically controllable gobos, separately shaping outer perimeters of each of said first, second and third light beams based on said first, second and third electronic files; and

a combiner that combines said first, second and third light beams with shaped outer perimeters into a composite light beam.

45. An apparatus as in claim **44**, wherein said first, second and third light beams are each in different primary colors.

* * * * *

EXHIBIT T

US007181112B2

(12) **United States Patent**
Harris

(10) **Patent No.: US 7,181,112 B2**
(45) **Date of Patent: Feb. 20, 2007**

(54) **THREE COLOR DIGITAL GOBO SYSTEM**

(75) Inventor: **Jeremiah J. Harris**, Las Vegas, NV (US)

(73) Assignee: **Production Resource Group, L.L.C.**, New Windsor, NY (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **11/386,194**

(22) Filed: **Mar. 21, 2006**

(65) **Prior Publication Data**

US 2006/0177185 A1 Aug. 10, 2006

Related U.S. Application Data

(63) Continuation of application No. 10/995,612, filed on Nov. 22, 2004, now Pat. No. 7,020,370, which is a continuation of application No. 10/616,481, filed on Jul. 8, 2003, now Pat. No. 6,823,119, which is a continuation of application No. 09/771,953, filed on Jan. 29, 2001, now Pat. No. 6,588,944.

(51) **Int. Cl.**

G02B 6/44	(2006.01)
G02B 26/00	(2006.01)
G09F 13/00	(2006.01)

(52) **U.S. Cl.** **385/100**; 385/115; 385/116; 385/88; 385/147; 385/901; 359/291; 362/232; 362/551; 362/556

(58) **Field of Classification Search** 385/88, 385/89, 92, 49, 115, 116, 14, 147, 901, 37, 385/100; 359/291, 223, 224; 382/217, 220, 382/190; 348/241, 239, 246; 362/232, 551, 362/556, 293, 296

See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,053,765 A	10/1991	Sonehara et al.	......	340/815.31
5,774,174 A *	6/1998	Hardie	....	348/38
5,828,485 A	10/1998	Hewlett	....	339/291
5,940,204 A	8/1999	Hewlett	....	359/298
6,057,958 A	5/2000	Hunt	....	359/291
6,188,933 B1	2/2001	Hewlett et al.	....	700/19
6,208,087 B1	3/2001	Hughes et al.	....	315/291
6,256,136 B1	7/2001	Hunt	....	359/291
6,538,797 B1	3/2003	Hunt	....	359/291
6,588,944 B2	7/2003	Harris	....	385/88
6,617,792 B2	9/2003	Hughes et al.	....	315/32
6,823,119 B2	11/2004	Harris	....	385/100
6,891,656 B2	5/2005	Hunt	....	359/291
2002/0181070 A1 *	12/2002	Hewlett	....	359/291
2005/0100289 A1 *	5/2005	Harris	....	385/88
2006/0177185 A1 *	8/2006	Harris	....	385/100

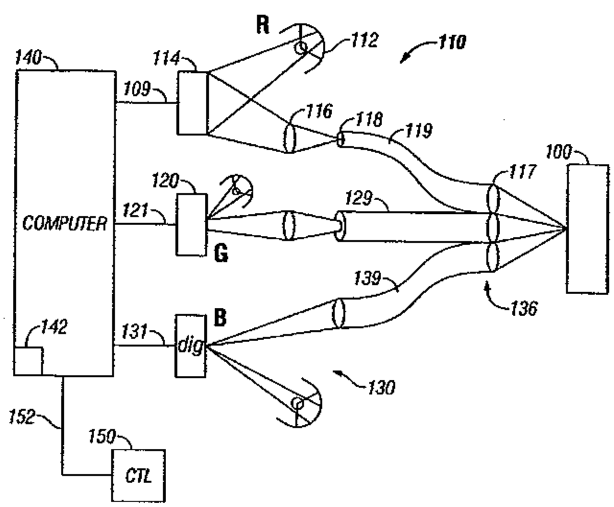
* cited by examiner

Primary Examiner—Brian M. Healy
(74) *Attorney, Agent, or Firm*—Fish & Richardson P.C.

(57) **ABSTRACT**

A system of digitally controlling light output by producing separate control signals for different colors of light. The light is contained in an optical waveguide, either prior to shaping or after shaping. Each of the control signals is coupled to a digitally controlled device which controls the shape of the light output. The digital controlling device can be digital mirror devices, for example.

45 Claims, 2 Drawing Sheets



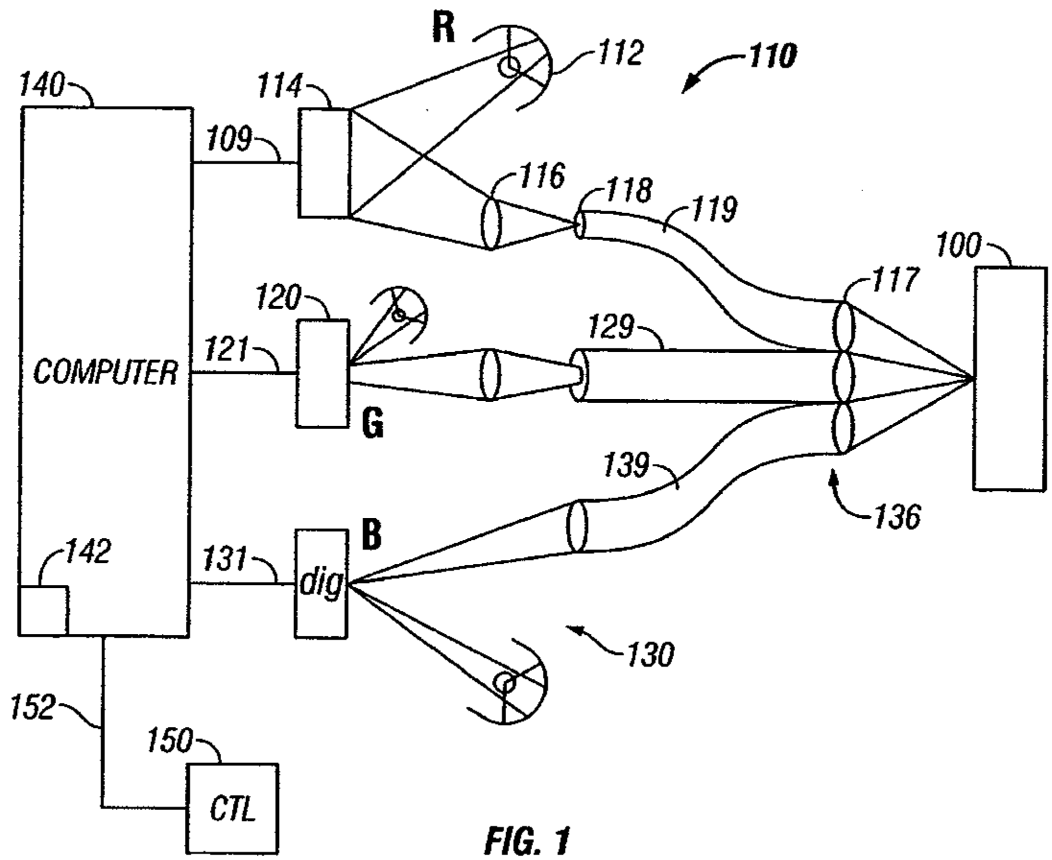

R
112
110
140
114
109
116
118
119
100
117
120
121
COMPUTER
129
G
139
136
142
131
dig
B
130
152
150
CTL

FIG. 1

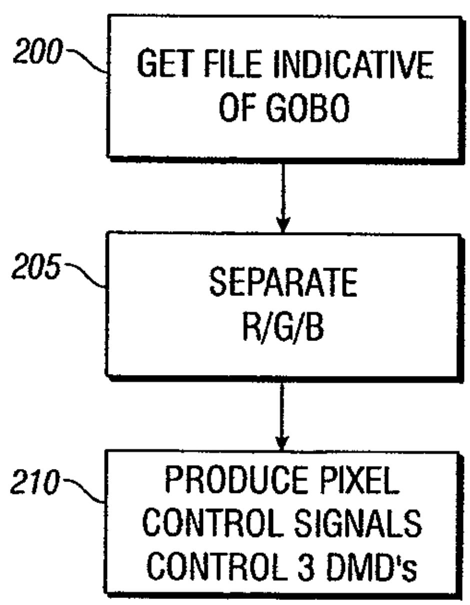

200
GET FILE INDICATIVE OF GOBO
205
SEPARATE R/G/B
210
PRODUCE PIXEL CONTROL SIGNALS CONTROL 3 DMD's

FIG. 2

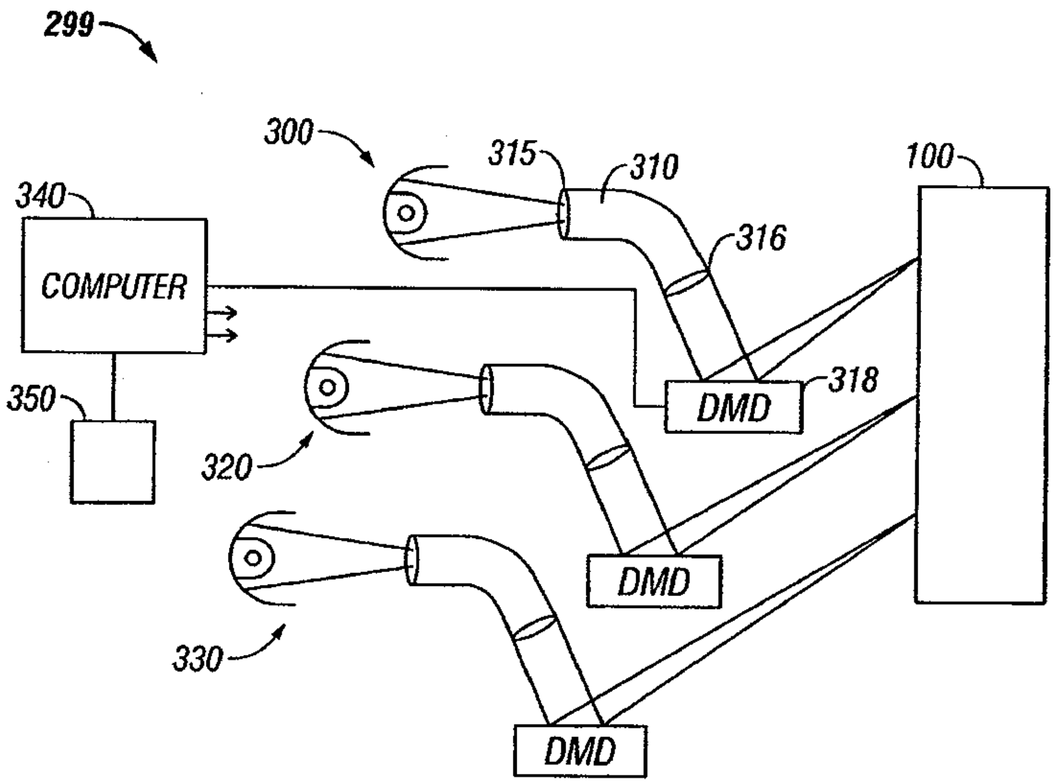

299
300
315
310
316
100
340
COMPUTER
318
DMD
350
320
DMD
330
DMD

FIG. 3

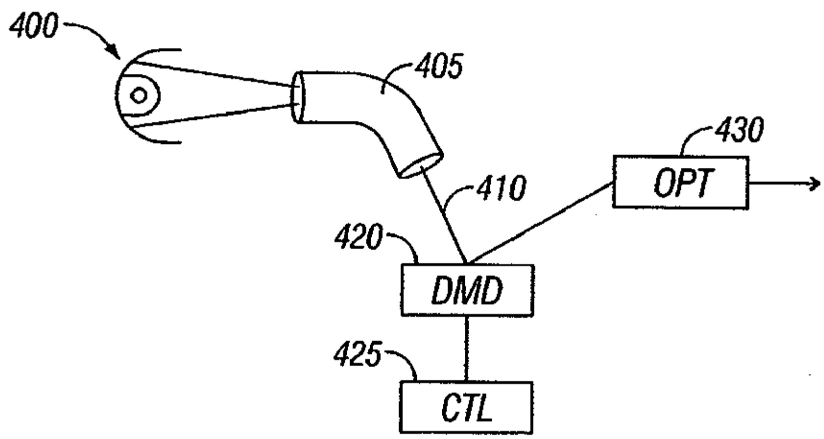

400
405
430
OPT
410
420
DMD
425
CTL

FIG. 4

THREE COLOR DIGITAL GOBO SYSTEM

CROSS-REFERENCE TO RELATED APPLICATIONS

This application is a continuation application of and claims priority to U.S. application Ser. No. 10/995,612, filed Nov. 22, 2004, now U.S. Pat. No. 7,020,370, which is a continuation of U.S. application Ser. No. 10/616,481, filed Jul. 8, 2003, now U.S. Pat. No. 6,823,119, which is a continuation of U.S. application Ser. No. 09/771,953, filed Jan. 29, 2001, now U.S. Pat. No. 6,588,944.

BACKGROUND

The U.S. Pat. No. 5,940,204 has suggested using a digital device to shape the contour and outlines of light that is projected through a high-intensity projector. Such a system may be used, for example, for stage lighting in theatrical and concert events. The Icon M™, available from Light and Sound Design, Ltd; Birmingham, England, uses this technique.

Different patents owned by Light and Sound Design, Ltd. suggest that the digital gobo should be formed from either a digital mirror, or from any other pixel level controllable digital device.

Cogent Light of Los Angeles, Calif. has technology that allows packaging a high intensity light beam into a form that allows it to be placed into a light waveguide, e.g., a fiber optic cable.

SUMMARY

The present application teaches a system of packaging light into a light waveguide such as a fiber optic cable, and adjusting the shape of the light using a digitally controllable, pixel level controllable light shaping element, such as a digital mirror device (DMD), available from Texas Instruments.

In one embodiment, the system controls and produces high-intensity light output using three separate digital gobo devices. The digital gobo devices can be separately controlled such that each digital gobo device receives information indicative of shaping a separate primary color. The primary colors are handled separately, and/or combined at the object of the high-intensity light output.

BRIEF DESCRIPTION OF THE DRAWINGS

These and other aspects will now be described in detail with reference to the accounts, wherein:

FIG. 1 is a block diagram of a three color version of the system.

FIG. 2 shows a flowchart of operation of the controlling process for the digital gobo's in FIG. 1.

FIG. 3 shows a 3 DMD solution using three optical pipes.

FIG. 4 for shows a single DMD solution.

DETAILED DESCRIPTION

Details of a lighting instrument using a digital gobo are described in many patents owned by Light and Sound Design Ltd and the basic features are also present in Light and Sound Design's Icon M™ lighting fixture. The system described herein may use any of these basic features including details of computer-controlled cooling, and optics.

A block diagram of the basic system is shown in FIG. 1. An object of lighting **100** is shown. This object may be a stage, or may be any other object which is conventionally by a high-intensity lighting device. The high-intensity lighting device may be, for example, a lighting device which produces more than 100 watts of lighting output, preferably more than 500 watts of lighting output. Devices of this type conventionally use a spotlight with a special high intensity bulb for producing the desired illumination effect.

In FIG. 1, three separate lighting units are formed. Each lighting unit is responsible for producing light of a separate primary color. The primary colors can be red, green and blue for additive colors, and cyan, magenta and yellow for subtractive coloration.

Each of the lighting units **110**, **120** and **130** are formed of similar structure. The lighting unit **110** includes a light source **112** which produces light of a specified primary color, here red. The lighting unit **110** may produce red coloration, or may include a white light with a red filter, or may even produce pure white light which is later filtered. The light from source **112** is applied to digital gobo device **114**. The digital gobo device **114** may be a digital mirror device available from Texas Instruments. Alternatively, the digital mirror device can be some other digitally controllable, pixel level controllable optical device such as, but not limited to, a grating light valve. The digital gobo device **114** is a controlling computer **140** which runs a specified program **142**. A controller **150** may be remote from the computer **140**, and connected to the computer by a line **152**. For example, the computer **140** may be within a separate lighting fixture along with the lighting elements **110**, **120** and **130**, and a remote central controller **150** may be a lighting control console.

The light output from the digital mirror device **114** is focused by an optics assembly **116**, and focused to the input end **118** of an optical waveguide **119**. The optical waveguide **119** may be, for example, a fiber-optic device including single or multiple fibers. The light input at end **119** is output at end **117**, and coupled towards the object **100**. Analogously, the other lighting unit **120** focuses its light onto a fiber-optic device **129**, and the lighting device **130** focuses its light onto a fiber-optic device **139**. Each of the lights may have different characteristics, i.e. they may have different coloration. The output of the three fiber-optic devices **119**, **129** and **139** are bundled together at area **136**, and are pointed towards the object of lighting **100**.

In this way, a number of advantages may be obtained. First, brighter light and different kinds of control may be obtained since the system disclosed herein uses three separate light sources. Moreover, better control over the digital gobo may be obtained since red; green and blue are separately controlled. Less flickering may be obtained, and more brightness, as compared with a system that uses only one DMD. Still a system that uses only one DMD is contemplated as described herein.

Different modifications on this system are possible. Other optical waveguides besides a fiber-optic pipe may be used in this system. Moreover, the optical filter which changes each of these separate light components to a separate light characteristic may be located after the digital mirror, e.g. as part of the optics assembly **116**, or on the input end of the fiber-optic device **118**.

The system is controlled according to the flowchart of FIG. 2. At **200**, a file indicative of a shaping of the light, e.g. a gobo to be used, is obtained. This file may be, for example, of the format described in U.S. Pat. No. 6,057,958. Of course, any file format can be used to define the gobo. The

definition can be monochrome, gray scale, or full color (three different colors). At **205**, the file is changed to an image, and separated into its primary color components. In the example given herein, the primary color components may include red, green and blue. Hence the file is separated into red, green and blue components. Such separation is conventional in video processing, and produces three separate signals. These three separate signals will eventually be used as the three separate controlling signals **109**, **121** and **131** respectively driving the red green and blue subassemblies. The control of the three separate digital mirror devices is carried out at **210**.

FIG. **3** shows an alternative embodiment which uses a similar concept. In the FIG. **3** embodiment, light is first launched from a light source **300** directly into a fiber-optic cable **310**. In this embodiment, the optics are shown as **315**, and are formed directly on the input end of the fiber-optic cable **310**. Light is launched into the fiber-optic cable, and hence may be focused and or colored by the optics **315**. Of course, this system may also use the separate optics shown as **116** in the FIG. **1** embodiment. Light is output on the output in **316** of the fiber-optic cable **310**, and coupled to a digital mirror device **318** which shapes the light and reflects it towards the object **100**.

The above has described a first channel shown as **299**. A separate second channel **320** produces a similar light alteration for the second aspect of light, while a third channel **330** produces a separate output for the third aspect of light; where the aspects can be colors. Each of the digital mirror devices may be controlled by the computer shown as **340** which may be controlled from a remote console **350**.

While the above has described control using three separate colors, it should be understood that two separate colors could also alternatively be used. Moreover, while the above describes the different aspects of light which are separately controlled being colors, it should be understood that any different aspect of shaping the beam of light could be separately controlled. For example, one alternative might use different intensity lights, each of which are separately controlled to produce some other kind of effect.

Another embodiment is shown in FIG. **4**. In this embodiment, a single DMD solution is shown. Light from the light **400** is immediately launched into an optical waveguide, e.g. fiber **405**. The fiber can be located in any configuration. It produces its light output **410** at the area of DMD **420**. As conventional, the DMD is controlled by a controller **425**. An optical assembly **430** receives the light from the DMD, and transmits it towards the object of illumination. The optical element **430** may include a color changing element therein, or multiple color changing elements, in order to produce full-color output. For example, the optical element **430** may include a spinning Red/Green/Blue filter which spins in synchronism with the changing of patterns on the DMD.

Although only a few embodiments have been disclosed in detail above, other modifications are possible. All such modifications are intended to be encompassed within the following claims, in which:

What is claimed is:

1. A method, comprising: obtaining an electronic file representative of a color gobo shape;

dividing said electronic file into a first part representative of a first part of the color gobo shape, and into a second part representative of a second part of the color gobo shape, where the first part and second parts collectively represent a complete color gobo shape; and

using said first part to control a first part of shaping a light according to the color gobo shape, and using said second part to control a second part of shaping the light according to the color gobo shape.

2. A method as in claim **1**, wherein said first part is a first color component of the light shaping by the color gobo shape, and said second part is a second color component of the light shaping by the color gobo shape.

3. A method as in claim **1**, further comprising dividing said electronic file into an additional third part representative of a third part of the color gobo shape.

4. A method as in claim **3**, wherein said first part is a first color component, said second part is a second color component, and said third part is a third color component of the light shaping by the color gobo shape.

5. A method as in claim **4**, wherein said colors components are red, green and blue respectively.

6. A method as in claim **4**, wherein said color components are cyan magenta and yellow respectively.

7. A method as in claim **1**, wherein said using comprises digitally controlling shaping of the light.

8. A method as in claim **1**, wherein said using comprises digitally controlling shaping the light in separate digitally controllable pixel level light shape altering devices.

9. A method as in claim **1**, wherein said obtaining an electronic file comprises using a computer to process a digital file which is stored in said computer.

10. A method as in claim **8**, further comprising using said digitally controllable pixel light shape altering devices to shape at least one beam of light.

11. A method, comprising:

forming first and second electronic files, respectively indicative of first and second component parts, which first and second component parts collectively represent a color gobo that defines shaping of an outer perimeter of a beam of light, and that defines a color within the shaped outer perimeter; and

using both said first and second component parts to form the color gobo in the specified shape with the shaped outer perimeter, and with the defined color within the shaped outer perimeter, further comprising forming a third electronic file indicative of a third component part, and where said first, second and third component parts collectively represent the color gobo, and wherein said using comprises using all of the first, second and third component parts to form the color gobo.

12. A method as in claim **11**, wherein said using comprises applying the first component part to a first digitally controllable light shape altering device, applying the second component part to a second digitally controllable light shape altering device.

13. A method as in claim **12**, further comprising applying a first light beam to said first digital light shape altering device, applying a second light beam to said second digital light shape altering device, and combining outputs of said first and second light shape altering devices to form the color gobo.

14. A method as in claim **11**, wherein said first and second component parts represent different color components of the color gobo.

15. A method, comprising:

forming first and second electronic files, respectively indicative of first and second component parts, which first and second component parts collectively represent a color gobo that defines shaping of an outer perimeter of a beam of light, and that defines a color within the shaped outer perimeter; and

using both said first and second component parts to form the color gobo in the specified shape with the shaped

outer perimeter, and with the defined color within the shaped outer perimeter, wherein said forming comprises obtaining an electronic file indicative of a color gobo, and separating said electronic file into said first and second component parts.

16. A method, comprising:

separately shaping outer perimeters of each of first and second separate light beams; and

combining said first and second light beams with shaped outer perimeters into a composite light beam, wherein said combining comprises using a fiber-optic device to combine said light beam.

17. A method, comprising:

separately shaping outer perimeters of each of first and second separate light beams; and

combining said first and second light beams with shaped outer perimeters into a composite light beam, wherein said shaping comprises shaping said outer perimeters according to first and second component files, which first and second component files collectively represent a color gobo that shapes the outer perimeter of a beam of light and colors a section within said shaped outer perimeter.

18. A method as in claim 17, wherein said shaping comprises obtaining a file indicative of a color gobo, and separating said file into said first and second component files.

19. A method as in claim 18, wherein said first and second component files respectively represent primary color portions of the color gobo.

20. A method as in claim 16, further comprising using a first light source to create said first light beam and using a second light source to create said second light beam.

21. A method, comprising:

obtaining a color gobo electronic file representing a color gobo;

using said color gobo electronic file to form first, second and third electronic files, respectively indicative of first, second and third component parts, which first, second and third component parts collectively represent a color gobo that defines a shape of an outer perimeter of a beam of light, and that defines at least one color within the shaped outer perimeter;

obtaining first, second and third light beams;

separately shaping outer perimeters of each of said first, second and third light beams; and

combining said first, second and third light beams with shaped outer perimeters into a composite light beam.

22. A method as in claim 21, wherein said first, second and third light beams are each in different primary colors.

23. An apparatus, comprising:

a computer, obtaining an electronic file representative of a color gobo shape, and dividing said electronic file into a first part representative of a first part of the color gobo shape, and into a second part representative of a second part of the color gobo shape, where the first part and second parts collectively represent a complete color gobo shape; and

an optical part, using said first part to control a first part of shaping a light according to the color gobo shape, and using said second part to control a second part of shaping the light according to the color gobo shape.

24. An apparatus as in claim 23, wherein said first part is a first color component of the light shaping by the color gobo shape, and said second part is a second color component of the light shaping by the color gobo shape.

25. An apparatus as in claim 23, wherein said computer further divides said electronic file into an additional third part representative of a third part of the color gobo shape.

26. An apparatus as in claim 25, wherein said first part is a first color component, said second part is a second color component, and said third part is a third color component of the light shaping by the color gobo shape.

27. An apparatus as in claim 26, wherein said color components are red, green and blue respectively.

28. An apparatus as in claim 26, wherein said color components are cyan magenta and yellow respectively.

29. An apparatus as in claim 23, wherein said optical part includes a first digitally controllable light shape altering device, controlled by said first part, and a second digitally controllable light shape altering device, controlled by said second part.

30. An apparatus as in claim 25, wherein said optical part includes a first digitally controllable light shape altering device, controlled by said first part, and a second digitally controllable light shape altering device, controlled by said second part and a third digitally controllable light shape altering device, controlled by said third part.

31. An apparatus as in claim 30, further comprising three light sources, producing light in each of three primary colors.

32. An apparatus, comprising:

sa computer, operating to form first and second electronic files, respectively indicative of first and second component parts, which first and second component parts collectively represent a color gobo that defines shaping of an outer perimeter of a beam of light, and that defines a color within the shaped outer perimeter; and

an optical system, receiving said first and second electronic files, and forming the color gobo in the specified shape with the shaped outer perimeter, and with the defined color within the shaped outer perimeter based on said first and second optical files.

33. An apparatus as in claim 32, wherein said computer further forms a third electronic file indicative of a third component part, and where said first, second and third component parts collectively represent the color gobo, and wherein said is responsive to each of the first, second and third electronic files to form the color gobo.

34. An apparatus as in claim 32, wherein said optical system comprises a first digital light shape altering device receiving said first electronic file, and a second digital light shape altering device receiving the second electronic file.

35. An apparatus as in claim 34, wherein said digital light shape altering devices are digital mirror devices.

36. An apparatus as in claim 34, further comprising a first light source, applying a first light beam to said first digital light shape altering device, and a second light source applying a second light beam to said second digital light shape altering device, and an optical combiner, combining outputs of said first and second light shape altering devices to form the color gobo.

37. An apparatus as in claim 36, wherein said first light source produces a first color light, and said second light source produces a second color light.

38. An apparatus as in claim 32, wherein said computer processes an electronic file indicative of a color gobo to form said first and second component parts representative of primary color portions within said electronic file.

39. An apparatus, comprising: first and second light sources, producing first and second separate light beams;

first and second electronically controllable gobo devices, each separately shaping outer perimeters of each of first and second separate light beams; and

a combiner, combining said first and second light beams with shaped outer perimeters into a composite light beam.

40. An apparatus as in claim **39**, further comprising a third light source, forming a third light beam, and a third electronically controllable gobo device, separately shaping said outer perimeter of said third light beam, and wherein said combiner combines said first, second and third light beams.

41. An apparatus as in claim **39**, wherein said first and second light beams are of different colors.

42. An apparatus as in claim **39**, wherein said combiner comprises a fiber-optic device.

43. An apparatus as in claim **39**, wherein said first and second light sources produce light of different primary colors.

44. An apparatus, comprising:

a computer that stores a color gobo electronic file representing a color gobo and operates using said color gobo electronic file to form first, second and third electronic files, respectively indicative of first, second and third component parts, which first, second and third component parts collectively represent a color gobo that defines a shape of an outer perimeter of a beam of light, and that defines at least one color within the shaped outer perimeter;

first, second and third light sources, respectively producing first, second and third light beams;

first, second and third electronically controllable gobos, separately shaping outer perimeters of each of said first, second and third light beams based on said first, second and third electronic files; and

a combiner that combines said first, second and third light beams with shaped outer perimeters into a composite light beam.

45. An apparatus as in claim **44**, wherein said first, second and third light beams are each in different primary colors.

* * * * *